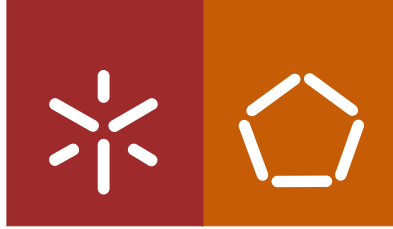


Universidade do Minho
Escola de Engenharia

António Pedro Cunha Martins da Silva

**Serviços + Perto: Prestação de Serviços ao
Cidadão Sobre Dispositivos Móveis**



Universidade do Minho

Escola de Engenharia

António Pedro Cunha Martins da Silva

Serviços + Perto: Prestação de Serviços ao Cidadão Sobre Dispositivos Móveis

Dissertação de Mestrado
Mestrado Integrado em Engenharia Biomédica

Trabalho realizado sob a orientação do
Professor Jorge Gustavo Rocha

Outubro de 2012

É AUTORIZADA A REPRODUÇÃO PARCIAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE;

Universidade do Minho, ____/____/____

Assinatura: _____

Agradecimentos

Na realização desta dissertação de mestrado não posso deixar de expressar os meus sinceros agradecimentos a todas as pessoas que contribuíram para o desenvolvimento e conclusão da mesma.

Em primeiro lugar gostaria de agradecer ao meu orientador, Professor Jorge Gustavo Rocha, pela oportunidade que me proporcionou, pela disponibilidade, atenção e motivação transmitidas, pelo bom aconselhamento a todos os níveis, pelos esclarecimentos prestados e pela confiança no meu trabalho.

À minha amiga Sara Alves agradeço pela grande ajuda que me deu na elaboração desta dissertação.

Agradeço aos meus amigos Janete Alves e João Pedro Ribeiro pela companhia e boa disposição ao longo de todo o meu percurso académico.

Agradeço os meus amigos Rui Silva, José Santos, Eliane Fernandes e Sebastião Cunha pela convivência e tempo partilhado ao longo da elaboração desta dissertação de mestrado, pela preciosa ajuda e disponibilidade sempre que foi necessário.

Agradeço também à minha família, com especial destaque aos meus pais e irmã por todo o apoio que sempre me deram, o incentivo e meios que me proporcionaram para a formação em toda a minha vida.

Por fim, agradeço à minha namorada Andrea Ribeiro pela paciência, respeito, dedicação e por me acompanhar nesta longa jornada que foi árdua para ambos.

Resumo

Numa era de grandes evoluções tecnológicas e de fácil acesso a todo o tipo de informação a nível global, gerou-se uma oportunidade há muito idealizada para que fossem libertados dados adquiridos pelos governos para uso livre, pela e para a comunidade em geral. No entanto, o acesso aos dados por parte de um cidadão por si só pode não ter grande impacto social. Posto isto, a utilização livre dos dados transformando-os em informação, seja de forma aberta ou comercial, por parte de indivíduos ou equipas de desenvolvimento especializadas poderá fornecer ao cidadão todo um leque de serviços até então inexistentes.

Assim, o principal objetivo deste trabalho passa por disponibilizar uma aplicação móvel que se torne uma mais-valia para os cidadãos, auxiliando-os na resposta a necessidades inerentes aos serviços públicos e privados mais procurados.

De forma a concretizar o objetivo proposto, foi efetuado um levantamento do funcionamento real da estrutura a representar. Posto isto, verificou-se a necessidade de criação de um *back-office* para gestão e centralização da informação associada ao modelo originado.

Como resultado, foi criada uma aplicação para Android que servirá como primeira ferramenta de medida dos benefícios que a solução pretendida poderá levar à vida dos cidadãos. Espera-se ainda que a aplicação sirva de incentivo à criação de soluções do mesmo âmbito.

Os objetivos foram alcançados e brevemente os cidadãos poderão usar a aplicação criada, assim como terão acesso a todo o código fonte para melhorias que possam ser acrescentadas ao sistema.

Abstract

In an era of great technological advances and easy, global access to all kinds of information, there is now an opportunity, idealized long-ago, to release data acquired by the governments to the public, rendering it free to use by the general community. However, a single citizen accessing this data might not make any significant social impact; thus, freely using this data and turning it into information, whether openly or commercially, by individuals or specialized development teams, could provide citizens with a new range of services that did not exist before.

This way, the main objective of this work is to create a mobile application that can be an asset to citizens and help them deal with the inherent necessities of public and private services that are more commonly and often used. In order to achieve the proposed goal, we observed the inner workings of the structure to be represented; therefore, there was the need to create a back-office to manage and centralize the information associated with the devised model.

As a result, an application for the Android system was created which will serve as the first tool of measurement of the benefits that the proposed solution will bring to citizens. It is also expected that this application will function as an incentive to create solutions in the same spectrum.

The objectives were achieved and soon citizens will be able to use the application, as well as have access to the source code so that improvements can be made and features added to the system.

Índice

Agradecimentos	iii
Resumo.....	v
Abstract	vii
Notação e Terminologia	xi
Índice de Figuras	xiii
Índice de Tabelas.....	xv
1. Introdução	1
1.1. Motivações e Objetivos.....	1
1.2. Metodologia	2
1.3. Estrutura.....	3
2. Enquadramento.....	5
2.1. E-Government	5
2.2. AMA.....	6
2.3. Iniciativas e Aplicações.....	7
2.3.1. <i>SpotHero</i>	8
2.3.2. <i>Fixit!</i>	9
2.3.3. <i>Mi Parque – Little Village</i>	9
2.3.4. <i>Taxi Share - Chicago</i>	10
2.3.5. <i>wheredoesmymoneygo.org</i>	11
3. Arquitetura	13
3.1. Modelo de Dados	13
3.1.1. <i>Estrutura Real</i>	13
3.1.2. <i>Modelo Entidade-Relação</i>	14
3.2. Back Office.....	17
3.3. Aplicação para Dispositivos Móveis	18
3.3.1. <i>Interface</i>	19
3.3.2. <i>Resposta</i>	21
3.3.3. <i>Plataforma de Desenvolvimento</i>	21
3.3.4. <i>Online vs. Offline</i>	21
4. Implementação	23
4.1. Base de Dados	23
4.2. Desenvolvimento Web	26

4.2.1.	<i>ExtJS 4</i>	26
4.2.2.	<i>Model</i>	26
4.2.3.	<i>View</i>	27
4.2.4.	<i>Controller</i>	30
4.2.5.	<i>Server-side</i>	30
4.3.	Desenvolvimento para Android	31
4.3.1.	<i>Base de Dados</i>	32
4.3.2.	<i>Activities</i>	33
4.3.3.	<i>Permissões</i>	44
5.	Caso Prático de Utilização	45
5.1.	Fluxo da Informação.....	45
5.2.	Uso da Aplicação Android	49
6.	Conclusões e Trabalho Futuro.....	57
6.1.	Conclusões.....	57
6.2.	Trabalho Futuro.....	58
	Referências Bibliográficas	61

Notação e Terminologia

Notação

Ao longo do documento utiliza-se texto em itálico para palavras em língua estrangeira que, dada a sua universalidade não foram traduzidas. O texto a negrito é utilizado para realçar a importância de algumas palavras.

Acrónimos

ADT	<i>Android Developer Tools</i>
AMA	Agência para a Modernização Administrativa
BD	Base de Dados
BMS	Balcões Multisserviços
CAOP	Carta Administrativa Oficial de Portugal
CSS	<i>Cascading Style Sheets</i>
DAO	<i>Data Access Object</i>
EUA	Estados Unidos da América
HTML	<i>HyperText Markup Language</i>
IDE	<i>Integrated Development Environment</i>
IRN	Instituto dos Registos e Notariado
JSON	<i>JavaScript Object Notation</i>
LC	Loja do Cidadão
MVC	<i>Model-View-Controller</i>
OCTO	<i>Office of the Chief Technology Officer</i>
OSM	<i>Open Street Map</i>
PAC	Posto de Apoio ao Cidadão
PHP	<i>PHP: Hypertext Preprocessor</i>

RIA	<i>Rich Internet Application</i>
SGBD	Sistema Gestor de Base de Dados
SO	Sistema Operativo
SQL	<i>Structured Query Language</i>
WYSIWYG	<i>What You See Is What You Get</i>
XML	<i>eXtensible Markup Language</i>

Índice de Figuras

<i>Figura: 2.1 – Aplicação SpotHero</i>	<i>8</i>
<i>Figura: 2.2 – Aplicação Fixit!</i>	<i>9</i>
<i>Figura: 2.3 – Aplicação Mi Parque – Little Village</i>	<i>10</i>
<i>Figura: 2.4 – Aplicação Taxi Share</i>	<i>11</i>
<i>Figura: 2.5 – Website wheredoesmymoneygo.org</i>	<i>11</i>
<i>Figura: 3.1 – Modelo Entidade-Relação</i>	<i>16</i>
<i>Figura: 3.2 – Atributos de Entidades</i>	<i>17</i>
<i>Figura: 3.3 - Fluxo da Solução Mobile</i>	<i>19</i>
<i>Figura: 3.4 – Vista do Top</i>	<i>20</i>
<i>Figura: 3.5 – (a) Menus de Navegação (b) Pesquisa Direta</i>	<i>20</i>
<i>Figura: 4.1 – Modelo Relacional da Base de Dados</i>	<i>24</i>
<i>Figura: 4.2 - Vista Principal do Back Office</i>	<i>28</i>
<i>Figura: 4.3 – Lista de Serviços</i>	<i>29</i>
<i>Figura: 4.4 – Janela de Edição de um Serviço</i>	<i>29</i>
<i>Figura: 4.5 - Gestão de Activities</i>	<i>31</i>
<i>Figura: 4.6 – Fluxo da Aplicação Android</i>	<i>35</i>
<i>Figura: 4.7 – Layout personalizado para as Tabs</i>	<i>36</i>
<i>Figura: 4.8 – Layouts personalizados para os Favoritos (a) Loja (b) Serviços</i>	<i>37</i>
<i>Figura: 4.9 – Layout personalizado de um item da lista de Edifícios</i>	<i>39</i>
<i>Figura: 4.10 – (a) Marcador de um Edifício no Mapa (b) Layout personalizado do Título do Diálogo</i>	<i>42</i>
<i>Figura: 4.11 – (a) Vista de Satélite (b) Vista Normal</i>	<i>43</i>
<i>Figura: 5.1 – Janela de Inserção/Edição de Entidades</i>	<i>46</i>
<i>Figura: 5.2 – Janela de Inserção/Edição de Serviços</i>	<i>47</i>
<i>Figura: 5.3 – (a) Janela de vínculo entre Conteúdos e Serviços (b) Janela de Inserção/Edição de Conteúdos</i>	<i>47</i>
<i>Figura: 5.4 – Janela de vínculo entre Categorias e Serviços</i>	<i>48</i>
<i>Figura: 5.5 – Janela de vínculo entre Balcões e Serviços</i>	<i>49</i>
<i>Figura: 5.6 – Página de entrada na Aplicação (Sem Favoritos)</i>	<i>50</i>

<i>Figura: 5.7 – (a) Lista de Categorias (b) Lista de Edifícios</i>	<i>51</i>
<i>Figura: 5.8 – (a) Lista de Serviços por Categorias (b) Lista de Serviços por Edifícios...</i>	<i>52</i>
<i>Figura: 5.9 – Detalhes do Serviço Cartão de Cidadão</i>	<i>53</i>
<i>Figura: 5.10 – Mapa com os Edifícios que permitam efetuar o Pedido do Cartão de Cidadão</i>	<i>53</i>
<i>Figura: 5.11 - (a) Diálogo de Edifício (b) Escolha de Navegação (c) Rota no Navigation</i>	<i>54</i>
<i>Figura: 5.12 – Página de Entrada (Com Favoritos).....</i>	<i>55</i>

Índice de Tabelas

Tabela 1 – Restrições do Tipo Unique	25
Tabela 2 – Modelos e Stores	27
Tabela 3 – Tipos de Activities	34

1. Introdução

1.1. *Motivações e Objetivos*

No seu dia-a-dia, qualquer cidadão depara-se com a necessidade de realizar determinados serviços inerentes ao seu quotidiano. Uma problemática relacionada com qualquer serviço reside na falta de informação prévia à sua resolução. Esta informação pode ser de vários âmbitos, podendo o local, os pré-requisitos, o custo, entre outros, variar de serviço para serviço.

No entanto, nem todos os cidadãos estão informados devidamente, ou por in experiência ou por falta de meios de acesso ao conteúdo disponível, levando a que por diversas vezes (no caso de serviços presenciais) os cidadãos tenham de se deslocar mais que uma vez por falta de um documento. Outras situações indesejáveis podem ser relacionadas a deslocações desnecessárias que resultam de desconhecimento de horários, qualquer tipo de indisponibilidade ou até pela existência de outras alternativas mais próximas. Sendo que o tempo despendido na realização de um serviço é um aspeto determinante, é indispensável garantir que uma mais-valia informativa agilize todo o processo.

Posto isto, este trabalho tem como objetivo fundamental disponibilizar uma aplicação móvel que permita aos cidadãos o fácil acesso à informação sobre os serviços prestados nas lojas do cidadão em qualquer momento e qualquer lugar, seja enquanto espera pelo metro, se dirige para o trabalho, durante o pequeno-almoço, etc. Tendo em conta as potencialidades dos dispositivos móveis, pretende-se também que os cidadãos possam através da aplicação contactar as entidades e/ou lojas associadas aos serviços a resolver.

1.2. Metodologia

Este projeto seguiu uma metodologia típica de desenvolvimento de *software*, na qual se exige um levantamento inicial de requisitos e funcionalidades. Este levantamento não foi apenas bibliográfico, pois era de grande interesse saber desde logo que tipo de aplicações e/ou serviços estão a ser disponibilizados aos cidadãos noutros países onde os movimentos a favor dos dados livres começou mais cedo, nomeadamente nos países anglo-saxónicos e noutros países europeus.

De seguida, foi feito um estudo e consequente aprendizagem de como se desenvolvem aplicações para as várias plataformas envolvidas, nomeadamente para a *Web* e plataformas móveis. Nesta fase, adicionalmente ao domínio do desenvolvimento das aplicações, foi necessário perceber todo o ciclo de vida das mesmas, principalmente o seu processo de disponibilização nos seus centros de distribuição.

Depois deste estudo inicial, partiu-se para a especificação da aplicação. Esta especificação resultou de um conjunto de requisitos anteriormente definidos, acordados com a Agência para a Modernização Administrativa (AMA). Foi também nesta fase que se projetou a arquitetura.

Com todo este trabalho de conceção realizado, seguiu-se a implementação prática do *back-office* capaz de proporcionar a agilização necessária à manutenção da informação inerente ao modelo de dados arquitetado. Testado este modelo, através de um protótipo desenvolvido para a *web*, passou-se à implementação da aplicação para os dispositivos móveis, neste caso concreto com o Sistema Operativo (SO) Android. Após a conclusão da aplicação, vários testes foram efetuados, os quais são exemplificados nesta dissertação (Capítulo 5) para verificar o bom funcionamento da plataforma criada.

Por fim, será disponibilizado o código fonte no portal svn.dados.gov.pt, assim como em outros locais de distribuição apropriados, para que esta aplicação possa ser estudada e enriquecida pela comunidade.

1.3. Estrutura

Esta dissertação está estruturada em 6 capítulos, encontrando-se estes organizados da seguinte forma:

Capítulo 1: Introdução

No primeiro capítulo efetua-se uma breve introdução aos principais conceitos, permitindo um enquadramento geral. A esta se segue a apresentação das motivações e são descritos os objetivos fundamentais do projeto, é exposta a metodologia utilizada, assim como a estrutura da dissertação.

Capítulo 2: Enquadramento

O segundo capítulo pretende apresentar, de forma mais pormenorizada, a situação atual contextualizada. É descrito o principal movimento que tem levado a uma massificação da disponibilização dos dados de forma livre por parte dos governos aos cidadãos, sendo referidos os principais impulsionadores do mesmo.

É feita uma apresentação da AMA, apontada a sua principal missão e referidos alguns dos seus esforços para uma maior aproximação entre os principais atores intervenientes nos serviços públicos.

Este capítulo aborda ainda as principais iniciativas que promovem o aparecimento de soluções como a pretendida por este projeto, referindo alguns casos de sucesso no panorama internacional. São ainda expostas várias aplicações que se enquadram no âmbito deste projeto, demonstrando as vantagens de uma disponibilização dos dados para serem usados pelos e para os cidadãos.

Capítulo 3: Arquitetura

Este capítulo contém uma descrição da arquitetura tida em conta para a realização deste projeto, sendo expostos o modelo de dados idealizado, o *back office* que suporta toda a estrutura e o *front-end* para o cidadão que resulta de uma aplicação para dispositivos móveis. São também referidas e justificadas todas as tomadas de decisão que foram feitas nesta fase de projeção.

É ainda neste terceiro capítulo que é descrito o ciclo de vida da informação, mostrando o processo que permita que os utilizadores possam dispor desta o mais rapidamente possível.

Capítulo 4: Implementação

No quarto capítulo, é apresentada a forma como foi feita a implementação da arquitetura. São referidas as plataformas escolhidas para o desenvolvimento quer da solução *web*, quer da aplicação para os terminais móveis.

É também neste capítulo que é exposta a implementação do modelo de dados e descrito o processo de replicação da informação nos dispositivos móveis. Esta replicação será feita de cada vez que a aplicação for atualizada, seja por funcionalidades ou por uma alteração considerada relevante no conteúdo. Esta alteração pode dever-se a vários fatores, como por exemplo um aumento significativo dos conteúdos ou uma inativação de qualquer loja, balcão ou serviço.

Capítulo 5: Caso Prático de Utilização

Neste capítulo é feita uma apresentação dos resultados obtidos através de um caso prático, mostrando um exemplo concreto de todo o ciclo de vida da informação, desde a sua introdução no *back-office* até à sua utilização por parte de um cidadão. Este ciclo será dividido em duas fases distintas: o fluxo da informação e um caso prático de utilização da aplicação num dispositivo móvel.

Capítulo 6: Conclusões e Trabalho Futuro

Por fim, o último capítulo da dissertação sintetiza as ilações tiradas no final do projeto elaborado, terminando com sugestões fundamentadas de trabalho futuro a realizar no seguimento da solução final obtida. Estas sugestões consistem primordialmente de funcionalidades a adicionar à aplicação para terminais móveis assim como da possibilidade de maior abrangência a diferentes plataformas.

2. Enquadramento

2.1. *E-Government*

O *E-Government*, de uma forma genérica, consiste na utilização das novas tecnologias para o auxílio da interação entre os governos e os cidadãos. A sua definição por parte do World Bank é dada por “*E-Government* refere-se à utilização, por parte de entidades governamentais, de sistemas de informação (tal como a Internet e a computação móvel) que têm a capacidade de transformar as suas relações com os cidadãos, o setor privado e outras entidades públicas. Esses sistemas podem ter vários fins: prestação de serviços públicos de uma forma mais eficaz, melhor interação com as empresas e a indústria, uma maior qualificação dos cidadãos através do acesso à informação, ou uma gestão governamental mais eficiente. Os benefícios da sua utilização passam por uma menor corrupção, maior transparência e conveniência e ainda uma redução de custos” (1).

A perceção geral da prestação de serviços públicos é associada a processos de longa espera e de resolução inconsistente. No entanto, os governos com um maior desenvolvimento tecnológico já vêm a mudar essa perspetiva há algum tempo (2).

O estudo efetuado em 2012 pelo departamento de assuntos económicos e sociais das Nações Unidas, denominado *E-Government for the People*, destaca que a maioria dos países está a migrar para um modelo centralizado num único portal, a partir do qual os cidadãos podem aceder a todos os serviços públicos independentemente da entidade pública que os providencia, aumentando assim a sua eficiência (3).

A acompanhar a massificação dos dispositivos móveis, surgiu nos últimos anos o conceito de *m-Government*. Este consiste num complemento ao *E-Government*, através da utilização dos *smartphones* e *tablets* como plataformas de distribuição e acesso a serviços e informações com interesse para os cidadãos (1).

2.2. AMA

Desde Maio de 2007, a responsabilidade pela elaboração e avaliação do *E-Government* em Portugal recai sobre a Agência para a Modernização Administrativa (5). Esta agência “é o instituto público integrado na administração indireta do Estado que tem a missão de operacionalizar as iniciativas de modernização e impulsionar a participação e o envolvimento dos diferentes atores, instituições e responsáveis” (4).

Das principais tarefas inerentes à atuação da AMA podem-se destacar o incentivo à modernização da prestação e distribuição dos serviços públicos, a responsabilidade pela gestão e desenvolvimento da rede nacional de postos de atendimento e ainda a criação e manutenção de portais públicos interativos (4, 5).

Tendo em conta a área de ação contemplada pelas tarefas e objetivos traçados para a AMA, existem vários projetos sob a sua alçada que são de extrema importância para o desenvolvimento e para a modernização do *E-Government* em Portugal.

Portal do Cidadão

O Portal do Cidadão é o canal de distribuição, através da Internet, de informação e conteúdos adjacentes aos serviços prestados por entidades públicas e privadas inerentes aos interesses da vida de qualquer cidadão. O seu principal objetivo passa por proporcionar aos cidadãos uma forma cómoda e eficaz de aceder aos serviços públicos, sem que para isso tenham de se deslocar.

Dados.Gov e SVN.Gov

No seguimento da disponibilização em massa dos dados produzidos pelas entidades públicas a nível global, a AMA criou o portal **Dados.Gov**, onde a informação produzida pela Administração Pública se encontra centralizada e disponível para a utilização livre por parte dos cidadãos. Adicionalmente foi criado ainda o portal **SVN.Gov**, através do qual é possível o acesso ao código fonte das plataformas eletrónicas de domínio público.

2.3. *Iniciativas e Aplicações*

A evolução e maturação do *E-Government* tem despoletado, por parte das entidades responsáveis, o interesse na organização de iniciativas que suscitem o interesse dos agentes com capacidade para produzir soluções que visem um enriquecimento da vida dos cidadãos. A disponibilização dos dados do interesse dos cidadãos por parte dos governos é um passo importante para o estreitamento da relação com os mesmos. No entanto, estes dados por si só, sem qualquer tratamento, não providenciam qualquer valor acrescentado ao quotidiano de cada um. Para tal, é fundamental que se promovam iniciativas que motivem as pessoas qualificadas a criar soluções que façam uso dos dados disponíveis para otimizar a sua utilização por parte dos cidadãos.

Em 2008, a reunião de esforços entre o OCTO (de *Office of the Chief Technology Officer*) do distrito de Columbia nos Estados Unidos da América (EUA) e a iStrategyLabs resultou num concurso denominado *Apps for Democracy*. Este concurso foi um enorme sucesso, dando origem a 47 (quarenta e sete) aplicações sobre a *Web*, o iOS e o Facebook (6). O sucesso alcançado por esta iniciativa pioneira originou, naturalmente, o aparecimento de outras semelhantes, das quais se podem destacar a *NYC Big Apps* em Nova Iorque, *DataSF App Contest* em San Francisco, a *CivicApps* em Portland e a *Apps for Metro Chicago* em Chicago (7,8,9). Aos concursos originários nos EUA, seguiram-se outros do mesmo âmbito por todo o mundo, como são os casos do *MashupAustralia* na Austrália, o *Apps4Italy* em Itália e o *BCNappsJam* em Espanha (7,9,10).

As iniciativas e concursos que vão sendo organizados apresentam várias vantagens, das quais se podem destacar o incentivo à progressão tecnológica, à criação de postos de trabalho, entre outros. No entanto, o objetivo principal passa pelas soluções criadas e disponibilizadas para uso público.

2.3.1. SpotHero

A aplicação SpotHero foi projetada com o intuito de resolver um problema corrente na vida dos cidadãos de Chicago, que passa pela dificuldade em encontrar um local de estacionamento adequado às necessidades dos mesmos. O SpotHero providencia mecanismos de pesquisa da existência de lugares disponíveis para estacionamento, possibilitando que os cidadãos saibam de antemão o quanto vão pagar e que têm o seu lugar garantido.

Como se pode observar pela Figura 2.1, o cidadão pode pesquisar pela localidade pretendida obtendo os locais disponíveis. Pode ainda verificar os detalhes e proceder à marcação que melhor se ajuste às suas necessidades. Por fim, tem acesso ao recibo resultante do processo de marcação, através do qual poderá aceder ao lugar em questão.

É de salientar ainda que esta aplicação foi a vencedora do concurso *Apps for Metro Chicago* (11).

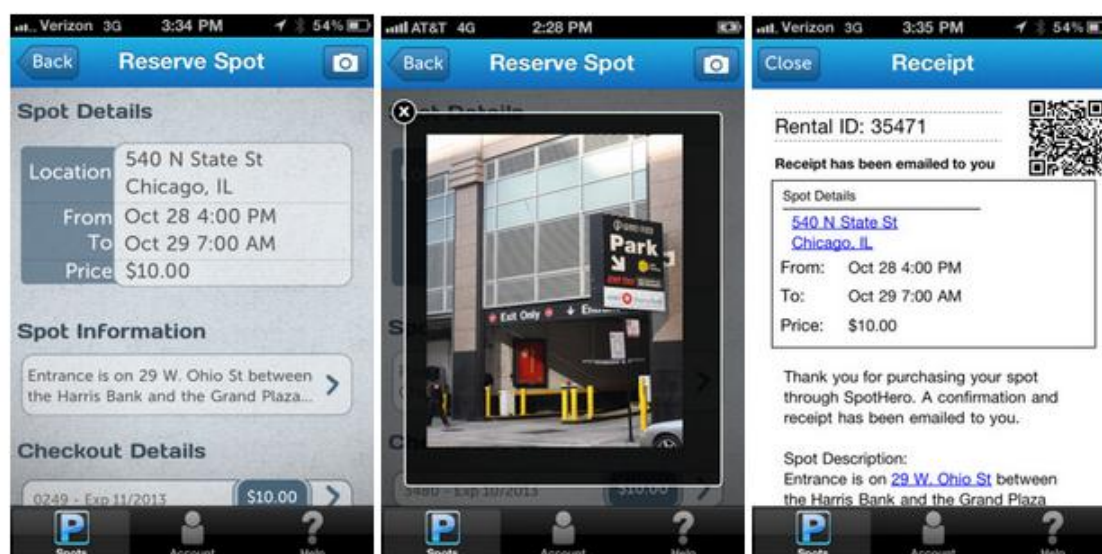


Figura: 2.1 – Aplicação SpotHero (24)

2.3.2. Fixit!

A Fixit! é uma aplicação que também surgiu como resposta ao concurso *Apps for Metro Chicago*. Esta solução pretende aproximar a colaboração entre as entidades responsáveis pela manutenção e limpeza da cidade de Chicago e os seus cidadãos. Como se pode verificar na Figura 2.2, todos os utilizadores podem reportar qualquer acontecimento que necessite de intervenção por parte das entidades responsáveis. Adicionalmente, os cidadãos poderão receber notificações de forma a poderem corroborar as situações reportadas, ou então retificar as mesmas.

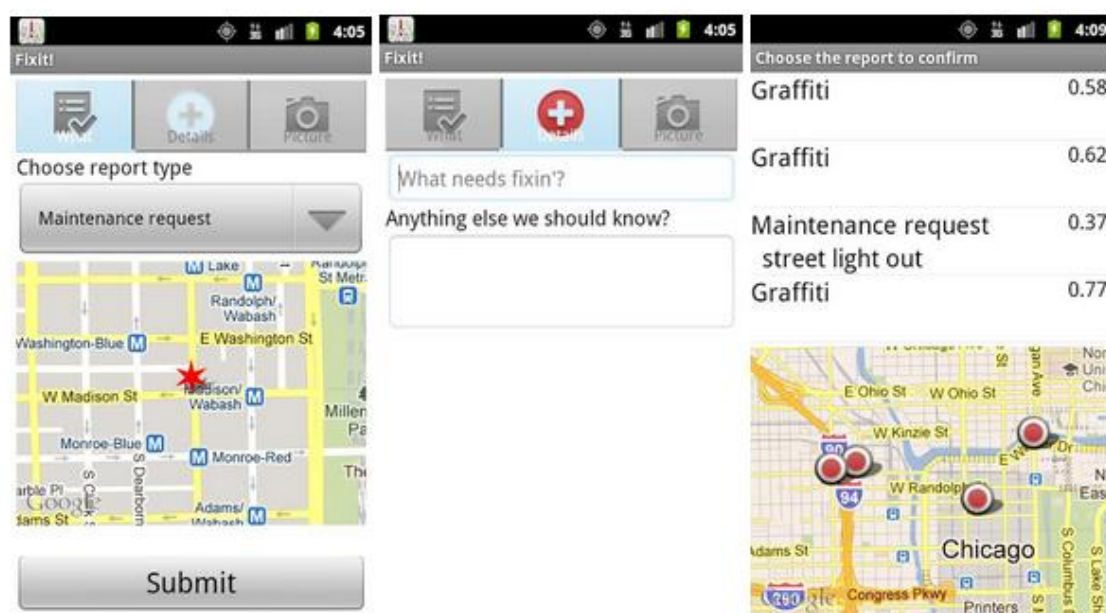


Figura: 2.2 – Aplicação Fixit! (25)

2.3.3. Mi Parque – Little Village

Mi Parque consiste numa solução que permite que os moradores da zona de Little Village (Chicago), na cidade de Chicago, possam contribuir para que o parque a que terão acesso a partir de 2014 seja usufruído por toda a comunidade envolvente (12). Através das aplicações para os dispositivos móveis ou então da aplicação construída sobre a *Web*, os interessados poderão contribuir com ideias, receber notícias sobre o tempo, os mapas ou até oportunidades de voluntariado que permitam ajudar todos os cidadãos de Little Village.

Tal como a SpotHero, também a Mi Parque consta das soluções galardoadas pelo concurso *Apps for Metro Chicago* (12).

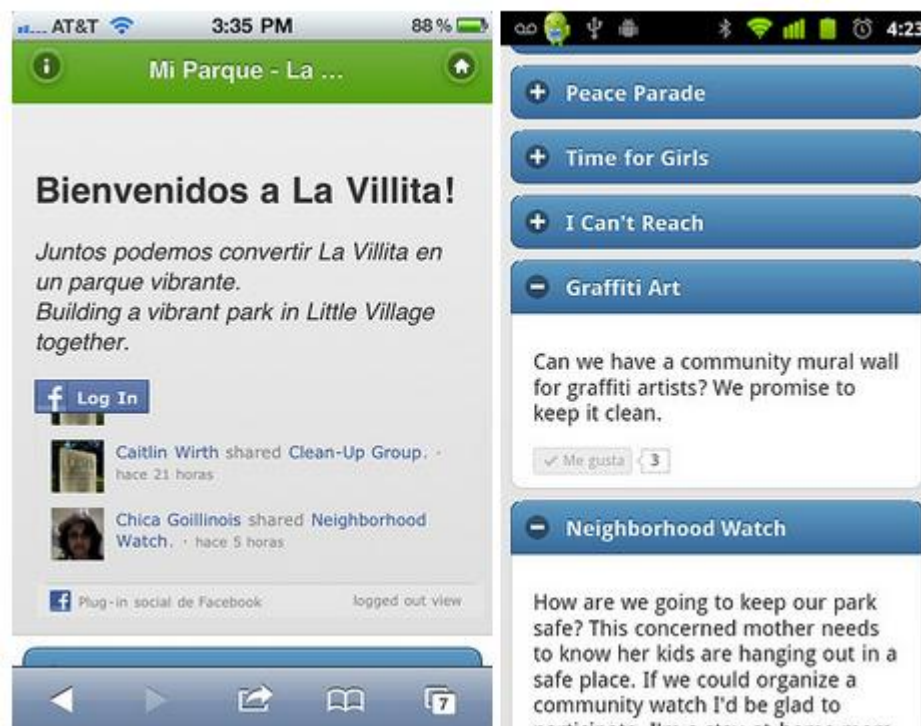


Figura: 2.3 – Aplicação Mi Parque – Little Village (12)

2.3.4. Taxi Share - Chicago

Em cidades de grande densidade populacional, a temática dos transportes é de extrema importância, tendo os táxis um papel fundamental para que os cidadãos se possam deslocar com a rapidez inerente às suas necessidades. Posto isto, ainda no âmbito da iniciativa *Apps for Metro Chicago*, surgiu a ideia de criar uma solução que permita que todos os utilizadores do táxi como meio de transporte, possam partilhar a sua viagem de forma a reduzir substancialmente os custos relacionados.

O funcionamento desta aplicação, como se pode verificar na Figura 2.4, prima pela simplicidade. Num primeiro instante o utilizador escolhe o destino pretendido, sendo brindado com a informação de outros utilizadores que se pretendam dirigir para o mesmo local. De seguida, é possível iniciar uma comunicação entre os utilizadores envolvidos e, em caso de concordância, proceder à chamada do táxi e partilha da viagem.

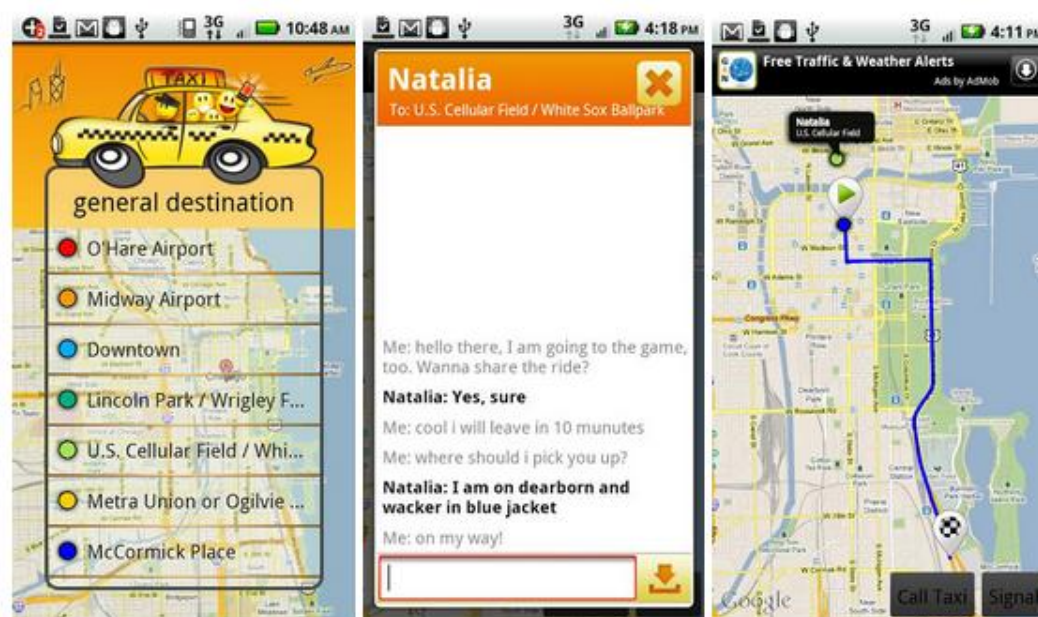


Figura: 2.4 – Aplicação Taxi Share (26)

2.3.5. *wheredoesmymoneygo.org*

Ainda como resultado de um concurso, mas neste caso no Reino Unido, surgiu um Website, através do qual os cidadãos podem ter uma noção do destino que é dado ao dinheiro proveniente dos seus impostos (13). Este projeto visa a promoção de uma maior transparência e envolvimento por parte dos cidadãos.

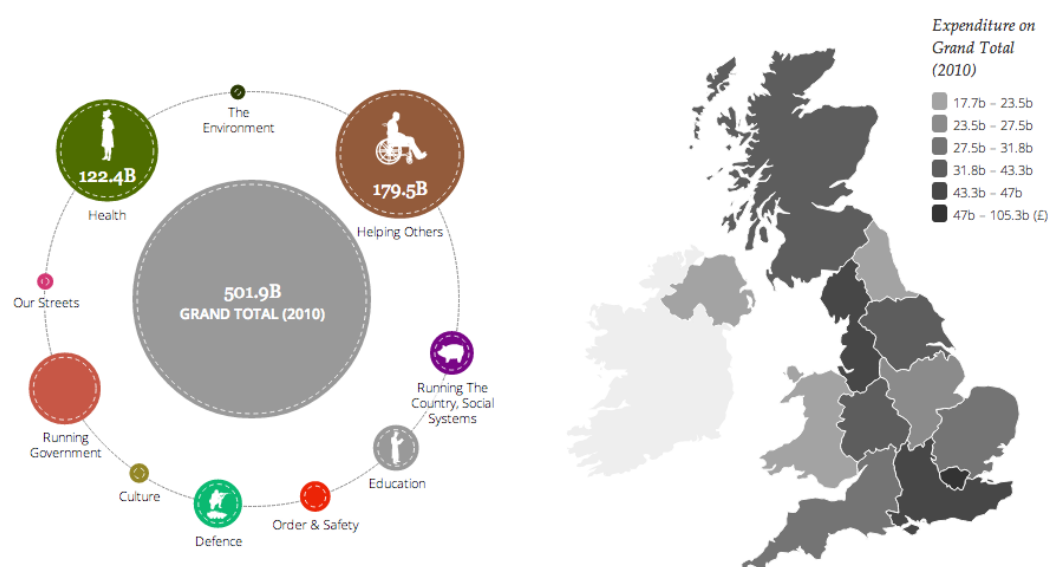


Figura: 2.5 – Website wheredoesmymoneygo.org (6)

3. Arquitetura

O portal do cidadão é o principal meio de distribuição de informação do interesse público. Por sua vez, as lojas do cidadão são os edifícios centralizados com um maior leque de serviços presenciais por parte de entidades públicas e privadas. Tendo em conta estas duas premissas foi elaborada a especificação de uma solução, em conjunto com a AMA, que organizasse a informação do portal do cidadão de forma concisa e a disponibilizasse aos cidadãos através de uma aplicação para os dispositivos móveis. Esta informação teria de ser orientada às necessidades inerentes de quem quisesse obter informações auxiliares à resolução de serviços relacionados com o quotidiano de qualquer cidadão.

Posto isto, foi necessário elaborar um modelo de dados escalável que permita responder da melhor forma às especificações requeridas. Os dados existentes no Portal do Cidadão não estão expostos de forma adequada à apresentação num dispositivo móvel. Por isso, foi necessária a criação de um modelo de dados que possibilitasse a edição e preparação desses mesmos dados para os dispositivos móveis. Sobre este modelo de dados assentam um *back-office* que suporta a edição e preparação dos dados e uma solução móvel para o utilizador, que pode passar por vários tipos de aplicações.

3.1. *Modelo de Dados*

A primeira fase da criação do modelo de dados consiste no levantamento do funcionamento real de toda a estrutura a representar, assim como dos fluxos inerentes a esse mesmo funcionamento. O sucesso a longo prazo do modelo a criar depende principalmente do trabalho de levantamento e da sua representação conceptual.

3.1.1. *Estrutura Real*

Quando um cidadão necessita solucionar qualquer problema através da resolução de um ou vários serviços, tem de se dirigir ao local apropriado para o efeito, sendo

este físico ou virtual. Dependendo do serviço, a sua resolução poderá necessitar de um atendimento presencial. No entanto, com as potencialidades conferidas ao cartão de cidadão surgiu uma tendência para a resolução *online* de vários serviços, como são os casos da alteração da morada, a consulta da lista de espera para cirurgia, criação de uma empresa, entre outros.

No que diz respeito à resolução presencial de um serviço, esta pode ser realizada em vários postos. Com o aparecimento das primeiras lojas do cidadão em Lisboa (Laranjeiras) e no Porto em 1999 (4), os cidadãos passaram a dispor de um vasto leque de serviços públicos e privados centralizados num único local. Devido ao sucesso das lojas do cidadão, foram criados também os Postos de Apoio ao Cidadão (PAC) que servem de extensão às mesmas em locais de menor densidade populacional. No entanto, tanto as entidades públicas como as privadas ainda mantêm um largo número de postos de atendimento individualizados por todo o país.

Em relação às entidades, todas elas prestam vários serviços aos cidadãos. Variando de entidade para entidade, estas podem ter balcões próprios dentro dos edifícios, como também podem ver os seus serviços ser prestados em postos de atendimento múltiplo, agora também denominados de Balcões Multisserviços (BMS). Os BMSs foram criados para otimizar e rentabilizar o aproveitamento do espaço nas lojas, agregando a prestação de vários serviços de entidades diferentes num só posto (4).

3.1.2. *Modelo Entidade-Relação*

Uma entidade pode ser descrita como “qualquer pessoa, objeto, evento ou conceito que seja indistinguível sobre a qual podemos armazenar informação” ou “qualquer objeto indistinguível que possa ser representado numa base de dados”. Posto isto, uma entidade é representada por uma tabela da Base de Dados (BD) no modelo relacional. Já a instância de uma entidade é representada pela ocorrência de um registo (linha) na tabela em questão. As entidades podem ainda ser de um tipo denominado por entidade associativa, tendo um significado especial já que estas são usadas para associar duas ou mais entidades cuja relação seja de muitas para muitas (∞) (14,15,16).

Desta forma, foram identificadas seis entidades fundamentais para o modelo da solução pretendida, sendo descritas de seguida:

- **Entidades** - As entidades representam qualquer entidade prestadora de serviços, seja esta pública ou privada.
- **Serviços** - Os serviços são o elemento base da solução, já que estes constituem o objetivo final dos cidadãos. Independentemente da informação que seja consultada, esta serve sempre como um meio para atingir um fim que passa pela resolução do serviço pretendido.
- **Conteúdos** - Estes representam informações específicas dos serviços, sendo que é através destes que os cidadãos podem suprir certas necessidades meramente informativas. Os conteúdos podem ser classificados em oito tipos diferentes: Descrição, Quem, Onde, Quando, Pré-Requisitos, Custos, Prazos, e Outras Informações.
- **Edifícios** - Estes representam qualquer edifício no qual uma ou mais entidades prestam um ou vários dos seus serviços. Estes edifícios podem ser por exemplo a Loja do Cidadão (LC) do Porto, o PAC de Guimarães, o posto dos Correios das Caldas das Taipas, entre outros.
- **Balcões** - Os balcões são uma entidade associativa, com o intuito de representar uma associação entre as entidades e os edifícios, seja esta na realidade de um posto físico ou não.
- **Relação entre Balcões e Serviços** - Esta é outra entidade associativa necessária, pois os serviços prestados em dois balcões da mesma entidade em edifícios diferentes podem ser os mesmos ou não, isto é, o balcão da Segurança Social na Loja do Cidadão das Laranjeiras pode prestar um menor ou maior número de serviços do que o balcão da mesma entidade na Loja do Cidadão de Braga.

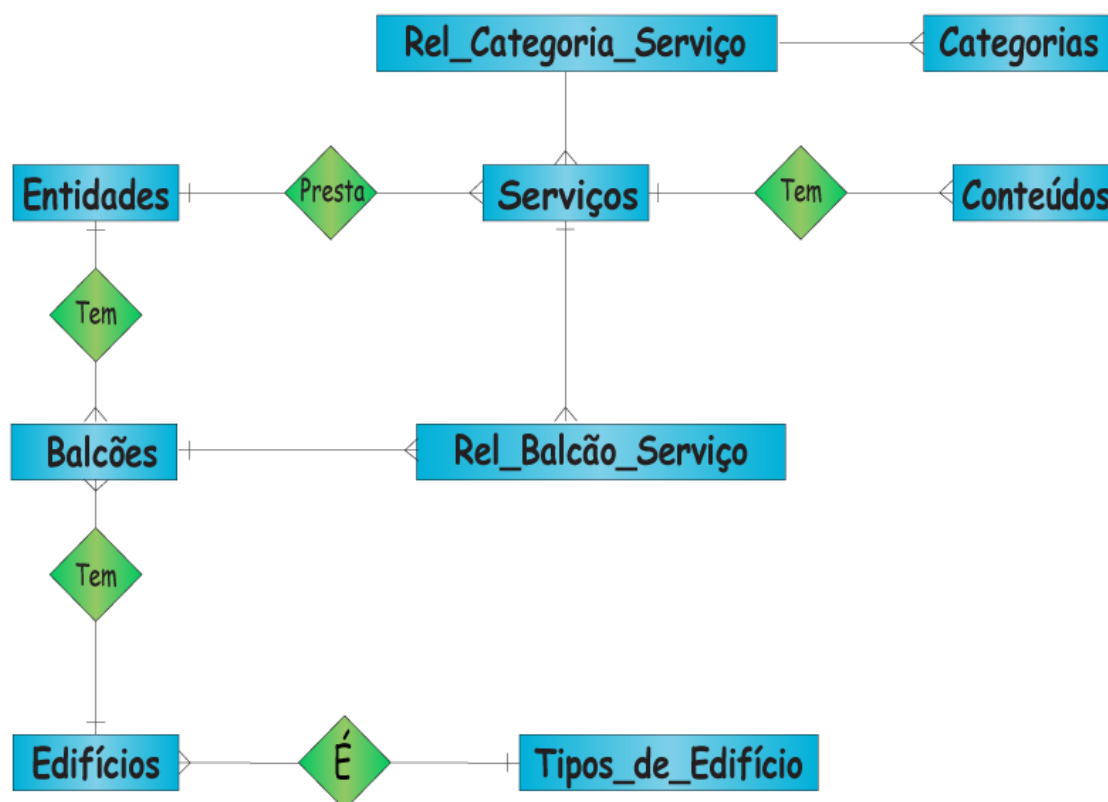


Figura: 3.1 – Modelo Entidade-Relação

Adicionalmente às entidades descritas foram ainda tidas em conta outras três, como se pode observar pela Figura 3.1. A entidade referente aos Tipos de Edifícios representa os diferentes tipos de edifício que podem existir, permitindo armazenar informação que seja comum a todos os edifícios do mesmo tipo. De forma análoga, a entidade das Categorias foi introduzida no modelo de dados para permitir categorizar os serviços e permitir o armazenamento de informação comum apenas a serviços da mesma categoria. Dado que a relação entre serviços e categorias é de muitos para muitos, foi criada ainda a Relação entre Categorias e Serviços.

Todas as entidades referidas possuem um conjunto de atributos que as descrevem, sendo estes representados pelas colunas de uma tabela da BD no modelo relacional. Posto isto, os atributos de cada entidade podem ser visualizados na Figura 3.2, estando estes representados pelas elipses.

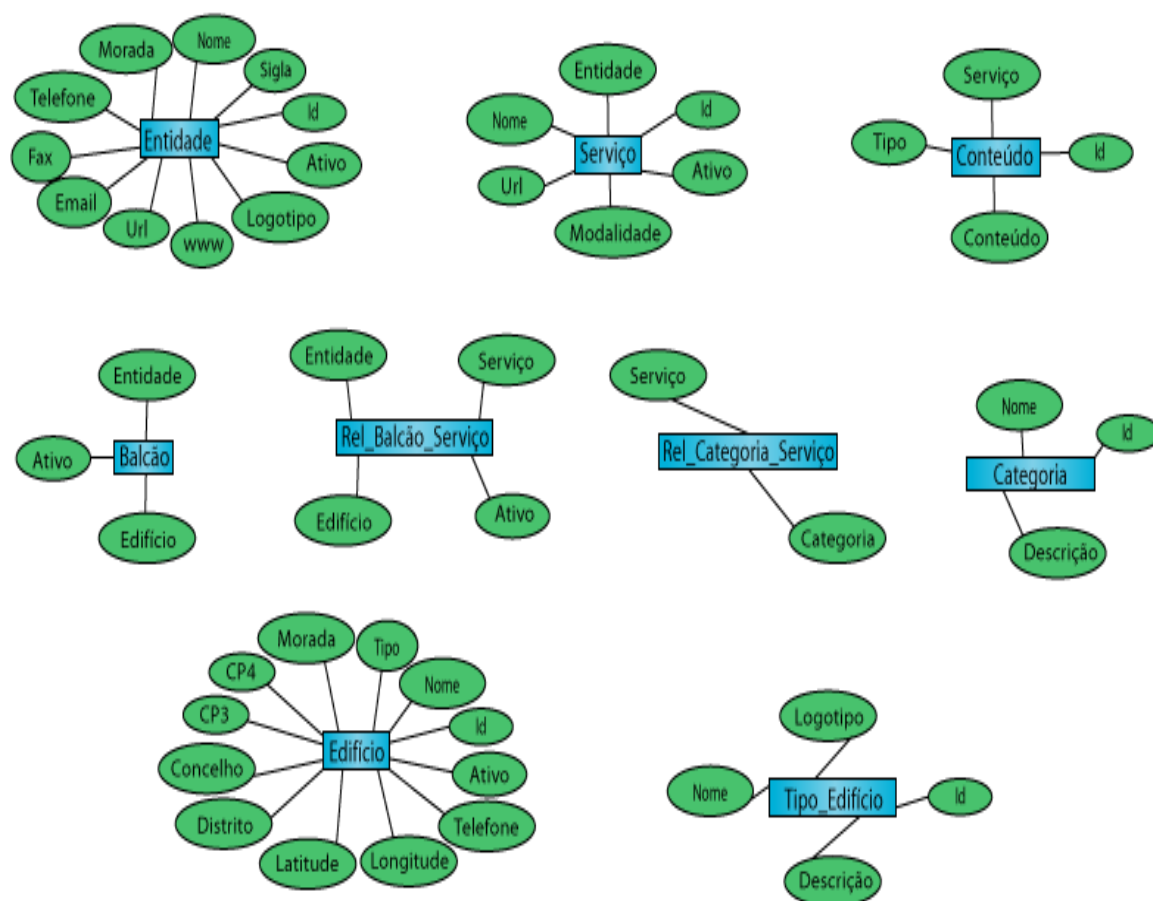


Figura: 3.2 – Atributos de Entidades

3.2. Back Office

A informação respeitante à solução pretendida é bastante abrangente e de vários âmbitos, encontrando-se os dados que a compõe dispersos por vários meios de comunicação e disponibilização. Posto isto, gerou-se a necessidade de centralizar estes dados tornando-os em informação que possa ser utilizada pela solução final e, consequentemente, pelos cidadãos.

Para que a informação a ser utilizada tenha o efeito pretendido é necessário que esta seja corretamente selecionada e articulada por alguém especializado, uma vez que a mesma assenta sobre as especificações descritas neste documento. Só assim será possível garantir a exatidão requerida. Adicionalmente, outras características determinantes para o bom funcionamento do sistema são a frequência com que a informação é atualizada e a rapidez com que estas atualizações se refletem na solução final.

A título de exemplo, “ (...) entre as 20:00h de sexta-feira, dia 28 de Setembro, e as 24:00h de domingo, dia 30 de Setembro (...) ” houve a indisponibilidade de serviços do Instituto dos Registos e Notariado (IRN). Notícias como esta são de extrema importância para que os cidadãos não se desloquem desnecessariamente.

Uma aplicação de *back-office* serve de suporte à gestão e manutenção de informação interna. Esta não é exposta aos utilizadores finais, sendo apenas utilizada com o intuito de tratar a informação de apoio à aplicação de *front-end*.

Tendo em conta toda a estrutura assente no funcionamento da prestação dos serviços públicos e privados subjacentes ao sistema em questão, é fundamental que a informação esteja centralizada numa BD acessível a qualquer pessoa que tenha a devida autorização, independentemente da sua localização ou SO utilizado. Assim sendo, optou-se pela construção de uma RIA (de *Rich Internet Application*) para a interação direta com a BD estando as duas alojadas no mesmo servidor. Com esta solução qualquer pessoa qualificada para o efeito pode atualizar a informação a partir do seu posto de trabalho, bastando para isso que faça uso do seu *web browser*.

3.3. *Aplicação para Dispositivos Móveis*

Inicialmente estava prevista uma aplicação para expor aos cidadãos informações sobre as lojas do cidadão, os balcões disponíveis e uma diversidade de estatísticas acumuladas nos últimos anos. É importante esta informação estar disponível, mas estes dados só têm valor quando resultam numa mais-valia para o cidadão, para além de satisfazer a sua mera curiosidade. Assim sendo, a projeção da solução final centra-se nas necessidades dos cidadãos e na forma como estes dados as podem satisfazer.

Desta forma, como se pode verificar na Figura 3.3, o fluxo da solução *mobile* assenta sobre uma variedade de vistas de interface com fim à obtenção de uma resposta que permita auxiliar na resolução da necessidade inicial, que passa de uma forma geral pela resolução de um serviço.

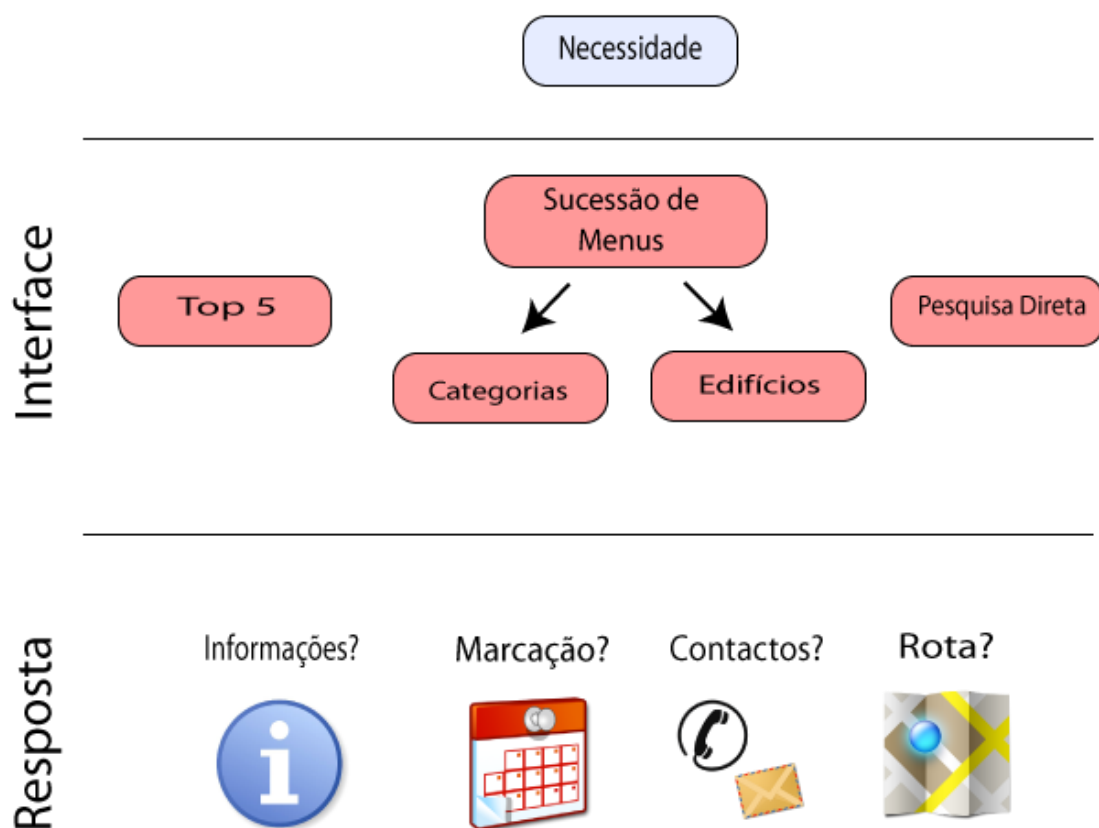


Figura: 3.3 - Fluxo da Solução *Mobile*

3.3.1. Interface

Para o sucesso da aplicação a desenvolver é fundamental que a interface seja o mais simples e perceptível possível, sendo que por mais funcionalidades que a aplicação possa ter, o utilizador necessita de aceder ao objetivo final de forma concisa e intuitiva. Assim sendo, existem três fatores determinantes que passam pela organização pessoal do utilizador, pelas características inerentes à estrutura de prestação dos serviços públicos e privados e, por fim, a necessidade de manter o utilizador a par de todas as notícias de relevo assinalável para o bom funcionamento do sistema.

Tendo em conta a organização pessoal de cada um, assim como as suas preferências, foi pensada uma página de entrada para a aplicação (Figura 3.4) que contemple um *top* de 5 (cinco) serviços previamente identificados pela AMA e um mecanismo de acesso aos edifícios e serviços da escolha do utilizador.



Figura: 3.4 – Vista do Top

O mecanismo principal da aplicação responsável pelo acesso à informação passa por dois tipos de abordagem direcionados ao serviço. Num primeiro plano pode ser apresentada ao utilizador uma sucessão de menus que se vão especializando até ao serviço final. Estes menus podem partir de uma vista por Categorias ou Edifícios dependendo das pretensões do cidadão (ver Figura 3.5 – (a)). De modo mais simplista e ágil, pode ser fornecida uma forma de pesquisa direta na qual o cidadão introduz o nome do serviço pretendido, obtendo instantaneamente os serviços filtrados pelo termo de pesquisa como é exemplificado na Figura 3.5 – (b).

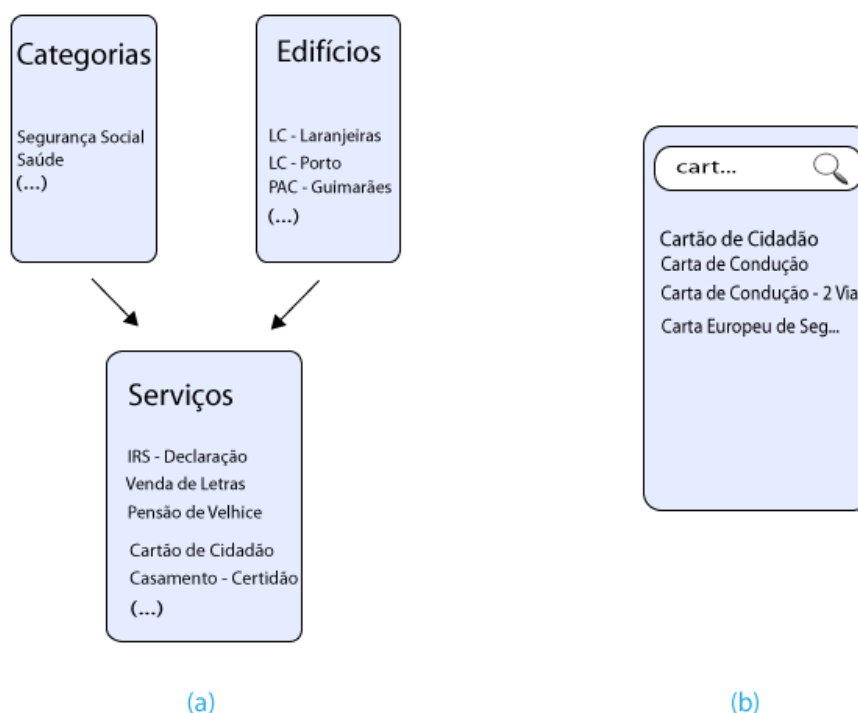


Figura: 3.5 – (a) Menus de Navegação | (b) Pesquisa Direta

3.3.2. *Resposta*

De forma generalizada, podem ser distinguidas duas componentes principais intrínsecas à solução pretendida. Num primeiro plano está uma componente com um teor maioritariamente informativo, dotando os cidadãos de um conhecimento enriquecido respeitante à resolução dos mais variados serviços. A outra componente passa pela resolução imediata de uma necessidade, seja pela disponibilização da rota para um determinado edifício, ou então pela possibilidade de contacto direto com o mesmo e/ou com a entidade responsável pelo serviço pretendido. Os contactos em questão podem ser por meio de uma chamada telefónica ou através do correio eletrónico.

3.3.3. *Plataforma de Desenvolvimento*

Atualmente em Portugal existem três sistemas operativos principais para os dispositivos móveis, sendo estes o *iOS* da *Apple*, o *Android* da *Google* e o *WindowsPhone* da *Microsoft*. Contudo, a escolha para a plataforma de desenvolvimento recaiu numa primeira fase sobre o *Android*, devido a um conjunto de fatores que o tornam mais apetecível para o efeito. Dos fatores de escolha podem-se destacar o custo de desenvolvimento e lançamento da aplicação nos respetivos centros de distribuição, assim como o *target* de utilizadores proporcionado pelo *Android* ser consideravelmente superior. De qualquer modo, o trabalho de preparação e gestão de dados poderá e deverá ser aproveitado para qualquer outro cliente móvel, tornando mais fácil e mais rápido esse desenvolvimento.

3.3.4. *Online vs. Offline*

Com a evolução da tecnologia e massificação da utilização dos dispositivos móveis (*smartphones* e *tablets*), tornou-se clara a necessidade do acesso à internet sem fios em qualquer lugar. No entanto, a panóplia de funcionalidades que a internet pode providenciar ainda não está tão acessível à maioria dos utilizadores quanto seria desejável, ora porque a maioria das redes Wi-Fi estão protegidas contra o uso de terceiros, ora pelos preços praticados pelas operadoras relativos aos tarifários de acesso às redes

3G e 4G. Como tal, foi necessária uma análise das funcionalidades que poderiam fazer uso (ou não) da internet:

- **Acesso à informação do modelo de dados** – No que diz respeito ao acesso à informação, faz todo o sentido que este seja feito independentemente do acesso à internet, já que sem esta a aplicação não teria qualquer propósito. É importante que os utilizadores possam fazer uso da aplicação sem que para isso sejam obrigados a gastar o seu tráfego das ligações de dados. Posto isto, terá de existir uma réplica no dispositivo móvel da informação presente na BD do servidor.
- **Mapas e Rotas** – Os mapas a utilizar numa fase inicial são os da Google pela sua maturidade e pela acessibilidade de implementação. No entanto, estes mapas apenas estão disponíveis *online* pelo que será necessária uma ligação de dados para que esta funcione. Para a obtenção das rotas será utilizada uma aplicação própria para o efeito, devido à interoperabilidade presente nas plataformas dos dispositivos móveis.

4. Implementação

No âmbito desta dissertação, para implementar o sistema arquitetado foi necessário recorrer a um SGBD (Sistema Gestor de Base de Dados) onde as informações inerentes ao modelo de dados pudessem persistir. A escolha para o SGBD a utilizar recaiu no PostgreSQL devido à sua enorme potencialidade que apresenta inúmeras vantagens. O facto de ser um projeto gratuito e aberto, aliado à sua conjugação com o PostGIS para armazenar dados georreferenciados fizeram do PostgreSQL uma escolha natural.

Em relação ao *back-office*, o *front-end* e o *back-end* (ou *server-side*) foram desenvolvidos com recurso a diferentes plataformas. Para o *front-end* foi utilizada a ExtJS 4, uma poderosa *framework* que assenta sobre o HTML (de *HyperText Markup Language*), CSS (de *Cascading Style Sheets*) e Javascript, permitindo também o uso de componentes com bastante maturidade e qualidade profissional sem a necessidade de grande esforço na manipulação do HTML e CSS. Já no que diz respeito ao *back-end*, é através deste que é efetuada a interação com a BD. Para a sua implementação foi usada a linguagem de programação PHP (PHP: Hypertext Preprocessor).

Por fim a aplicação para dispositivos móveis, tal como foi referido no Capítulo 3, foi desenvolvida para Android com recurso à linguagem de programação Java através da ferramenta aconselhada - *plugin* ADT (de *Android Developer Tools*) para o *Eclipse* (17).

4.1. Base de Dados

A base de dados criada em PostgreSQL pretende refletir todas as necessidades impostas pelo modelo de dados arquitetado, possuindo uma tabela por cada entidade presente no modelo Entidade-Relação descrito no Capítulo 3. Adicionalmente foram apenas criadas duas tabelas que permitem uma validação quando é introduzido um novo edifício. Estas tabelas contêm todos os distritos e concelhos existentes em Portugal continental e nas Ilhas de acordo com a Carta Administrativa Oficial de Portugal (CAOP).

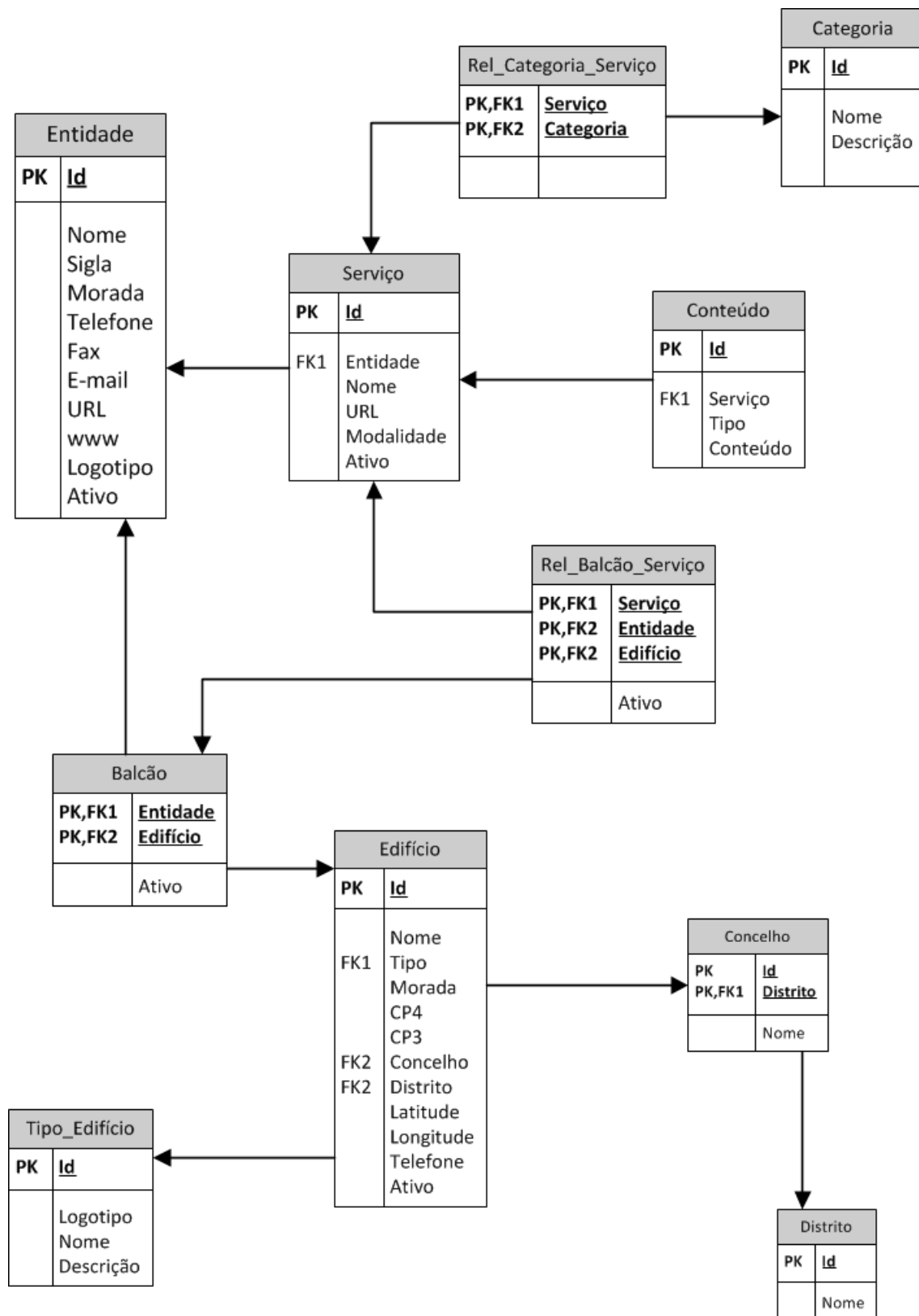


Figura: 4.1 – Modelo Relacional da Base de Dados

Como se pode observar pela Figura 4.1, as tabelas Entidade, Serviço, Edifício, Balcão e Rel_Balcão_Serviço possuem uma coluna **Ativo**. Este atributo foi criado para permitir alternar entre os estados Ativo e Desativo de qualquer registo contido nas tabelas em questão. Desta forma, para representar qualquer impossibilidade de realização de um serviço num determinado balcão, seria apenas necessário “desativar” o registo da tabela Rel_Balcão_Serviço. De forma análoga, no caso de um Balcão não prestar qualquer serviço durante um determinado período de tempo, poderia ser obtido o mesmo efeito desativando-o na tabela própria, aplicando-se o mesmo conceito nas restantes tabelas.

De notar ainda, que várias tabelas possuem restrições para garantir que não existe a possibilidade de serem introduzidos registos repetidos. Para tal, é usada a restrição **UNIQUE** em determinados atributos, como se pode verificar na Tabela 4.1:

Tabela 4.1 – Restrições do Tipo *Unique*

Tabela	Atributos
Entidade	Sigla, Nome
Serviço	Entidade, Nome, Modalidade
Conteúdo	Tipo, Serviço
Categoria	Nome
Edifício	Nome, Tipo
Tipo_Edifício	Nome

Para que a repetição não ocorra, a restrição **UNIQUE** não permite a existência de dois ou mais registos nas tabelas mencionadas, com os mesmos valores em simultâneo para os respetivos atributos.

4.2. Desenvolvimento Web

4.2.1. ExtJS 4

A ExtJS 4 é uma *framework* usada por um grande número de programadores por todo o mundo, que permite a construção de RIAs com uma grande compatibilidade entre os vários exploradores de internet. A variedade de componentes de UI (do inglês *User Interface*), aliada à capacidade de conjugação destes em vários *layouts* diferentes contidos em *containers*¹ para o efeito, torna esta *framework* numa poderosa ferramenta de desenvolvimento de aplicações que funcionam perfeitamente sobre a internet (18, 19).

Ao contrário das versões anteriores, esta quarta versão da ExtJS assenta sobre o padrão de desenvolvimento MVC (de *Model-View-Controller*). Este padrão permite uma maior versatilidade e separação de conceitos sendo o *Model* respeitante ao modelo de dados de uma aplicação, as *Views* as componentes gráficas que permitem a interação do utilizador com o anterior e, por fim, os *Controllers* responsáveis por toda a dinâmica e ações inerentes à conjugação das *Views* e dos *Models* (18, 19).

Outra grande vantagem desta *framework* reside no facto desta usar um paradigma de programação orientada a objetos, o que permite às aplicações uma flexibilidade e uma manutenção ao longo do tempo que de outra forma seria inatingível.

4.2.2. Model

O modelo² está separado em dois tipos de classes – *Models* e *Stores* - que se complementam, sendo estas criadas pela extensão das classes *Model* e *Store* pertencentes à *framework*. Os *Modelos* representam qualquer objeto passível de ser representado na aplicação, existindo por isso onze com o intuito de representar as entidades do modelo de dados. Adicionalmente, existe um modelo representativo dos itens presentes na árvore de navegação.

As *Stores* possuem funções muito semelhantes às de um DAO (do inglês *Data Access Object*), representando uma *cache* de objetos *Model* do lado do cliente e sendo

¹ Classe base para qualquer componente que possa conter outros componentes.

² Uso da palavra modelo para distinguir o *Model* (do MVC) da classe da *framework* *Ext.data.Model*

ainda as responsáveis pela comunicação com o lado do servidor. Como se pode verificar na Tabela 4.2, existe uma *Store* para cada Modelo.

Tabela 4.2 – *Models e Stores*

Models	Stores
<i>BackOffice</i>	<i>BackOffice</i>
<i>City</i>	<i>Cities</i>
<i>CrudBuilding</i>	<i>CrudBuildings</i>
<i>CrudBuildingType</i>	<i>CrudBuildingTypes</i>
<i>CrudCategory</i>	<i>CrudCategories</i>
<i>CrudCategoryService</i>	<i>CrudCategoryServices</i>
<i>CrudContent</i>	<i>CrudContents</i>
<i>CrudDesk</i>	<i>CrudDesks</i>
<i>CrudDeskService</i>	<i>CrudDeskServices</i>
<i>CrudEntity</i>	<i>CrudEntities</i>
<i>CrudService</i>	<i>CrudServices</i>
<i>State</i>	<i>States</i>

A comunicação com o lado do servidor é realizada com recurso a um *Proxy* do tipo Ajax, sendo este o mais indicado para situações em que o *front-end* e o *back-end* se encontram no mesmo domínio. Os dados transferidos nestas comunicações usam o padrão JSON (de *JavaScript Object Notation*) uma vez que a utilização deste é menos custosa quando comparada com a comunicação por XML (de *eXtensible Markup Language*).

4.2.3. View

As *views* são as componentes gráficas apresentadas aos utilizadores, que neste caso concreto serão os responsáveis pela introdução, atualização e manutenção da informação de todo o sistema. Devido à simplicidade e objetividade inerentes ao *back-office*, a vista principal da aplicação consiste de um menu de navegação do lado esquerdo a partir do qual se pode abrir num painel central um conjunto de *tabs*, sendo






cada uma referente à interação com a base de dados, tal como se pode observar pela Figura 4.2.

ID	Tipo	Nome	Morada	CP4	CP3	Telefone	Ativo
1	Loja do Cidadão	Águeda	Praça do Município	3754	500	707241107	Sim
33	Loja do Cidadão	Amares	Palácio da Justiça Largo do Município	4720	058	707241107	Sim
2	Loja do Cidadão	Aveiro	Rua Orlando Oliveira n.º 41 a 47	3800	4	707241107	Sim
3	Loja do Cidadão	Borba	Largo da Liberdade Lote 1	7150	102	707241107	Sim
4	Loja do Cidadão	Braga	Rua dos Granjinhos n.º 6	4704	575	707241107	Sim
5	Loja do Cidadão	Campo Maior	Rua Visconde Seabra n.º 2 e 2 A	7370	121	707241107	Sim
6	Loja do Cidadão	Cantanhede	Rua dos Bombeiros Voluntários n.º 7	3060	0	707241107	Sim
7	Loja do Cidadão	Castelo Branco	Rua do Saibreiro	6000	197	707241107	Sim
8	Loja do Cidadão	Coimbra	Avenida Central n.º 16 18 e 20 (à Avenida Fernão de Magalhães)	3000	607	707241107	Sim
9	Loja do Cidadão	Esmoriz	Avenida da Praia n.º 464 - Edifício EsmorizTur	3885	403	707241107	Sim
11	Loja do Cidadão	Faro	Mercado Municipal de Faro Largo Dr. Francisco Sá Carneiro	8000	151	707241107	Sim
10	Loja do Cidadão	F. E. Cinta	Av. Guerra Junqueiro	5180	104	707241107	Sim
12	Loja do Cidadão	Gondomar	Praça Luís de Camões São Cosme	4420	183	707241107	Sim
13	Loja do Cidadão	Guarda	Centro Comercial Vivaci Av. Bombeiros Voluntários Egitanenses n.º 5	6300	523	707241107	Sim
14	Loja do Cidadão	Laranjeiras	Rua Abranches Ferrão n.º 10	1600	1	707241107	Sim
16	Loja do Cidadão	Madeira	Avenida Arriaga Edifício Arriaga n.º 42 A	9000	64	291212200	Sim
35	Loja do Cidadão	Marvila	Centro Comercial Pingo Doce Bela Vista Avenida Santo Condestável Lo...	1900	806	707241107	Sim
17	Loja do Cidadão	Murça	Travessa do Tribunal 1.º Piso	5090	142	707241107	Sim
18	Loja do Cidadão	Odivelas	Odivelas Parque Loja 2048 Estrada de Paiã Casal do Troca	2675	626	707241107	Sim
19	Loja do Cidadão	Penafiel	Rua Joaquim de Araújo	4560	467	707241107	Sim
20	Loja do Cidadão	Pinhal Novo	Edifício do Mercado Municipal de Pinhal Novo Praça da Independência	2955	120	707241107	Sim

Figura: 4.2 - Vista Principal do Back Office

De uma forma geral, cada entidade do modelo de dados tem uma *Grid* e uma janela com um formulário a si associadas. Deste modo, devido a certas características partilhadas por estas *views* foram criadas duas classes abstratas para que pudessem ser estendidas e ajustadas a certas necessidades específicas sempre que necessário.

Como se pode verificar pela Figura 4.3, todas as *Grids* têm uma barra no topo e outra no fundo. A barra superior é dotada de um conjunto de botões cujas funcionalidades permitem adicionar, editar ou eliminar um ou vários registos, assim como uma atualização da própria lista. Adicionalmente, devido às entidades associativas na maior parte dos casos pode ainda haver a possibilidade de associação de outro tipo de registos. No que diz respeito à barra inferior, a mesma é composta por um mecanismo de paginação.

Serviços				
 Adicionar  Editar  Eliminar  Atualizar  Conteúdos				
ID	Entidade	Nome	Modalidade	Ativo
3	ACT	Direitos e Deveres no Trabalho	Informação	Sim
2	ADSE	Despesas de saúde da ADSE	Reembolso	Sim
1	ADSE	Cartão Europeu de Seguro de Doença	Pedido	Sim
65	ATA	Cartão de Contribuinte	Segunda Via	Sim
63	ATA	IRS	Declaração	Sim
64	ATA	Imposto Único de Circulação (IUC)	Pagamento	Sim
5	CGA	Abono de Família - Crianças e Jovens	Pedido	Sim
6	CGA	Contagem de Tempo de Ex-Subscritor	Pedido	Sim
7	CGA	Subsídio de Funeral	Pedido	Sim
4	CGA	Abono de Família Pré-Natal	Pedido	Sim
9	CGA	Pensão de Aposentação	Pedido	Sim
8	CGA	Pensão de Sobrevivência	Pedido	Sim
66	DGAI	Cadernos de Recenseamento Eleitoral	Consulta/Correção	Sim
10	DGAJ	Certificado de Registo Criminal	Pedido	Sim
11	DGAJ	Certificado de Contumácia	Pedido	Sim
12	DGC	Conflitos de Consumo	Reclamação	Sim
62	DGS	Centro de Saúde	Transferência	Sim
61	DGS	Centro de Saúde	Inscrição	Sim
59	DGS	Consulta no Centro de Saúde	Marcação	Sim
60	DGS	Lista de Espera - Cirurgia	Consulta	Sim
15	DGSS	Cartão Europeu de Seguro de Doença	Pedido	Sim

Page 1 of 3 Displaying 1 - 25 of 68

Figura: 4.3 – Lista de Serviços

A Figura 4.4 permite exemplificar uma janela de inserção/edição de um registo. As janelas também se assemelham na generalidade, tendo todas elas um botão para guardar, outro para cancelar a operação e um formulário de dados. Por sua vez, os formulários diferem entre eles dadas as diferenças intrínsecas de cada Modelo.

Editar Serviço

Nome:

Cartão de Cidadão

Modalidade:

Pedido

Descrição:

Entidade:

IRN

Portal do Cidadão:

http://www.portaldocidadao.pt/PORTAL/entidades/MJ/IRN/pt/SER_cartao+de+ciudad

Ativo:

☒ Sim
 ☐ Não

Guardar

Cancelar

Figura: 4.4 – Janela de Edição de um Serviço

4.2.4. *Controller*

Os *Controllers* são o elo de ligação entre os *Models* e as *Views*, estando responsáveis por todas as interações e fluxos da aplicação. Assim sendo, foram criados dois *Controllers* de características completamente distintas, sendo um responsável pelo funcionamento do menu e o outro voltado para as interações e ações inerentes aos Modelos e suas respectivas *Stores* e *Views*. O segundo *controller* referido é abstrato, tendo permitido a sua extensão por outros seis. Apesar de existirem onze Modelos respeitantes ao modelo de dados, existem apenas seis *Controllers* relacionados com estes. Em primeiro lugar os Distritos e Concelhos não podem ser adicionados, editados ou eliminados, e também porque os Conteúdos e certas entidades associativas não são tratados em *Controllers* próprios.

4.2.5. *Server-side*

O lado do servidor é o responsável pela comunicação com a BD e foi desenvolvido na linguagem de programação PHP, tendo-se seguido também neste caso o paradigma da programação orientada a objetos. A interação com o PostgreSQL é feita com recurso à camada de abstração de base de dados - MDB2.

Posto isto, foi criada uma classe abstrata denominada Base que implementa a ligação à BD assim como os métodos mais genéricos, um de leitura do conjunto de registos de uma tabela e outro para eliminar um ou vários registos de uma determinada tabela. As restantes onze classes, respeitantes às entidades do modelo de dados, estendem esta classe Base, implementando os métodos de inserção e edição de registos tendo em contas as suas próprias características. De notar ainda que as classes *Cities* e *States* relacionadas com os Concelhos e Distritos, respectivamente, apenas têm a sua própria implementação do método de ler registos, não possuindo qualquer capacidade de inserção ou edição.

4.3. *Desenvolvimento para Android*

O desenvolvimento de aplicações nativas para o Android é efetuado com recurso à linguagem de programação Java e a uma *framework* própria - Android SDK - escrita na mesma linguagem. É aconselhado o uso do plugin ADT para o Eclipse tornando este IDE (do inglês *Integrated Development Environment*) numa ferramenta capaz de suprir todas as necessidades inerentes ao desenvolvimento de uma aplicação para Android. É ainda possível utilizar emuladores para testar as aplicações durante a sua fase de desenvolvimento (20, 21).

Todas as *frameworks* de desenvolvimento de *software* usam componentes próprios, com os quais os programadores necessitam de se familiarizar para poderem criar aplicações a partir das mesmas (21). Desta forma, para a aplicação desenvolvida os conceitos de *Activity*, *Intent* e *View* tiveram de ser compreendidos.

De uma forma abrangente, uma *Activity* é um componente que representa uma página de ecrã inteiro dentro de uma aplicação. Quando o utilizador pretende efetuar uma determinada tarefa, geralmente passa por uma ou várias *Activities*. Para que tal aconteça, é necessário existir um mecanismo de comunicação sendo este providenciado pelos *Intents*. As *Activities* são geridas pelo próprio sistema e são colocadas numa pilha (do inglês *stack*). Sempre que uma *Activity* invoca outra através de um *Intent*, a nova é criada e colocada no topo da pilha para que o utilizador possa interagir com a mesma. É de referir ainda que uma *Activity* pode invocar *Activities* pertencentes a outras aplicações (20, 21).

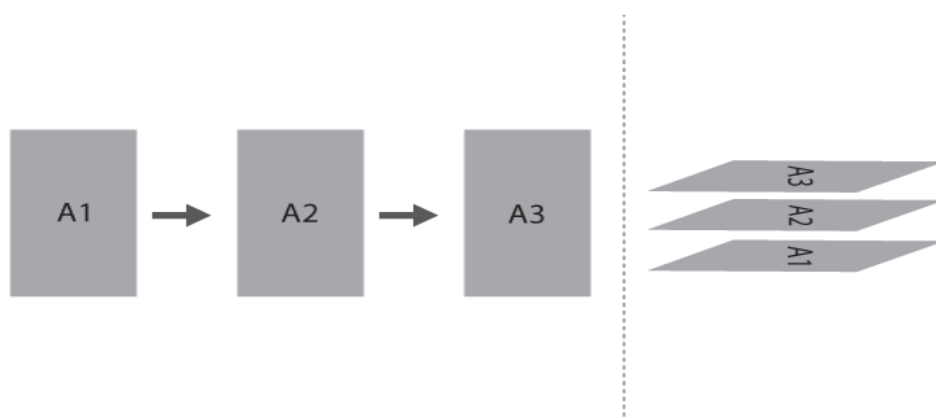


Figura: 4.5 - Gestão de *Activities*

Um *Intent* é o mecanismo usado para a comunicação entre as *Activities*, representando uma intenção de realizar uma determinada tarefa. Os *Intents* podem ser implícitos ou explícitos, ou seja, uma *Activity* pode invocar diretamente outra específica ou pode apenas comunicar a necessidade de realizar uma tarefa e, neste último caso, são expostas todas as *Activities* de aplicações instaladas no dispositivo que possam realizar a tarefa pretendida, permitindo assim que seja o utilizador a escolher.

As *Views* são os elementos que formam a interface gráfica que é apresentada aos utilizadores. Estes elementos estão organizados numa hierarquia. Ou seja, uma *View* (todo o conteúdo visível) de uma *Activity* é composta por um conjunto de *Views*, que por sua vez pode ainda ser composta pela combinação de outras *Views*, e assim sucessivamente. As *views* podem ser criadas de duas formas, diretamente no código da aplicação dentro de cada *Activity* ou em XML, sendo a segunda forma a aconselhada.

A *framework* disponibiliza um vasto leque de menus, sendo que o mais utilizado é o menu de opções. Este tipo de menu está associado a cada *Activity*, estando acessível através de uma tecla física para efeito, presente na grande maioria de dispositivos Android. Através do menu de opções o programador pode fornecer um vasto leque de funcionalidades relacionadas com a *Activity* à qual o menu pertence.

Existe ainda um ficheiro muito importante em cada aplicação - *AndroidManifest.xml*. Este ficheiro representa um manifesto da aplicação e é no seu conteúdo que estão presentes as listas de todas as *Activities* e Permissões que a aplicação necessita para funcionar.

4.3.1. Base de Dados

Tal como foi referido no Capítulo 3, o acesso aos dados por parte da aplicação terá de ser efetuado independentemente da existência, ou não, de uma ligação à internet. Posto isto, existe a necessidade de criar um processo de replicação da BD presente no servidor para os dispositivos, a ser executado na instalação da aplicação.

O SQLite é um motor de base de dados relacional e que já se encontra de raiz no Android. Uma BD em SQLite assenta num único ficheiro e a própria *framework* do Android disponibiliza classes próprias de auxílio ao seu uso. Desta forma foi criada uma

classe própria para lidar com todos os métodos de acesso à BD e a mesma possui uma classe privada que expande a `SQLiteOpenHelper`, permitindo a criação e atualização da BD quando a aplicação inicia. De notar que, o processo de criação da BD só é realizado na primeira vez que a aplicação é inicializada e o método de atualização só se realiza quando a versão da BD é alterada.

Para a criação da base de dados no dispositivo, sem que toda a informação tenha de ser descarregada da internet pela aplicação, existem duas possibilidades – utilizar uma BD `SQLite` já existente ou criar uma na primeira vez que a aplicação seja iniciada. Na prática, no primeiro caso seria criada uma BD vazia e o seu ficheiro seria substituído pelo final que iria no pacote da aplicação. No entanto existe uma grande desvantagem nesta solução que passa pelo facto de o ficheiro não poder ser movido mas apenas copiado, levando à duplicação do espaço ocupado. O segundo caso implica a execução dos comandos SQL (Structured Query Language) de criação e população da base de dados, o que também constitui uma desvantagem, já que o tempo de execução dos mesmos pode ser bastante custoso no arranque da aplicação. Neste caso concreto, não sendo o número de registos demasiado elevado, optou-se pela segunda opção.

Adicionalmente, foi criada na BD do dispositivo uma tabela extra – Favorito – que permite armazenar os edifícios e serviços favoritos de cada utilizador. Inicialmente esta tabela está vazia, sendo posteriormente introduzidos ou eliminados registos consoante as preferências de cada utilizador.

4.3.2. *Activities*

Especificadas todas as tarefas que o utilizador poderá efetuar na aplicação, foi necessário implementar as *Activities* e os fluxos que permitissem a realização dessas mesmas tarefas. Tal como se pode observar pela Figura 4.6, existem 10 (dez) *Activities* sendo as duas listas de serviços representadas instâncias diferentes da mesma *Activity*. Foi ainda criada outra *Activity* que não está representada na figura mas que serve para renderizar os detalhes de cada serviço, podendo a mesma existir através de uma ou mais instâncias contidas na *Activity* - Detalhes do Serviço.

Tendo em conta a estrutura e as *Views* inerentes a uma *Activity*, esta pode ser estendida de uma classe que já possui funcionalidades adicionais que conferem uma melhor usabilidade e aproveitamento por parte do programador. Na Tabela 4.3 estão presentes todas as *Activities* criadas e os seus respetivos tipos.

Tabela 4.3 – Tipos de *Activities*

Tipos de <i>Activity</i>	<i>Activities</i>
<i>Activity</i>	<i>BuildingDetailsActivity</i> , <i>ContactActivity</i> , <i>EntityDetailsActivity</i> , <i>TopActivity</i> , <i>WebViewActivity</i>
<i>TabActivity</i>	<i>MainActivity</i> , <i>ServiceDetailsActivity</i>
<i>ListActivity</i>	<i>BuildingListActivity</i> , <i>CategoriesActivity</i> , <i>ServiceListActivity</i>
<i>MapActivity</i>	<i>BuildingsMapActivity</i>

As *Activities* cuja apresentação não pressupõe nenhuma funcionalidade específica foram criadas pela extensão da classe *Activity*. Já as funcionalidades da *TabActivity* tornam-na uma ótima escolha quando existe uma necessidade de navegação pelo conteúdo através de *Tabs*. Por outro lado, todas as *Activities* criadas com o intuito de apresentar os seus conteúdos nas listas típicas devem estender a classe *ListActivity*. Por último, foi usada a classe *MapActivity* dadas as suas características especiais para lidar com os mapas da Google.

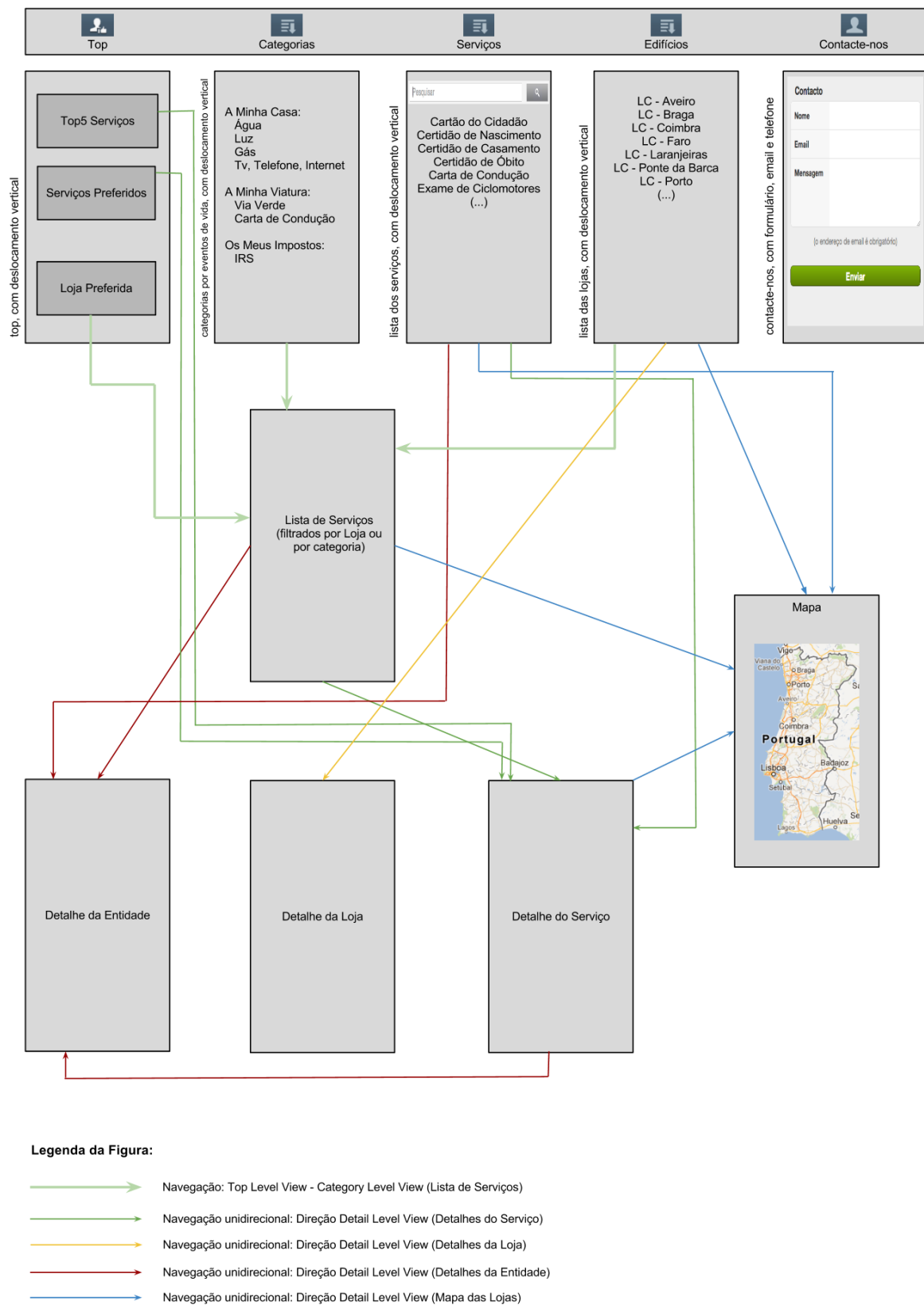


Figura: 4.6 – Fluxo da Aplicação Android

MainActivity

A MainActivity é responsável pelo início da aplicação, permitindo que o utilizador navegue por diferentes *Activities* contidas nos conteúdos das suas *Tabs*. É também na MainActivity que recai a responsabilidade de criar a BD na primeira vez que o utilizador inicia a aplicação. Sempre que é inicializada a MainActivity apresenta por defeito a página do top e dos favoritos.

O *layout* desta *Activity* é um *TabHost* que contém uma *HorizontalScrollView* onde está contido o *TabWidget*, uma *TextView* e um *FrameLayout*. A *HorizontalScrollView* serve para ser usada quando o número de *Tabs* não é facilmente apresentado no ecrã, de modo a que seja possível efetuar um efeito de arraste para alcançar as *Tabs* que estejam para lá dos limites do mesmo. Foi ainda criado um *layout* personalizado para cada *Tab* (ver Figura 4.7) que consiste apenas do texto a apresentar na mesma, mas que permite também que a sua cor seja alterada consoante o seu estado de selecionada ou não. O intuito da *TextView* é meramente estético já que apenas contem o fundo da cor da *Tab* selecionada. Por fim, o *FrameLayout* é o elemento onde serão colocados os conteúdos de cada *Tab*.

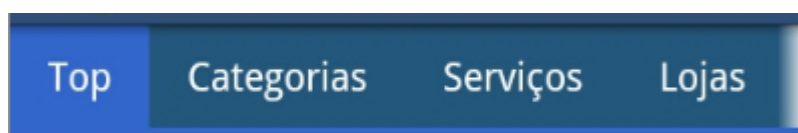


Figura: 4.7 – *Layout* personalizado para as *Tabs*

O menu desta *Activity* possui apenas uma funcionalidade que consiste em mostrar as informações sobre os criadores da aplicação.

TopActivity

A TopActivity como “página de entrada” apresenta o Top de 5 (cinco) serviços fornecido pela AMA e, quando existentes, as escolhas do utilizador como seus edifícios e serviços favoritos.

O *layout* criado para esta *Activity* é constituído por uma *ScrollView* que tem uma orientação vertical já que, dependendo da quantidade de conteúdos, pode ser demasiado extensa para os ecrãs de menor dimensão. Contidos nesta *ScrollView*, es-

tão presentes verticalmente duas `HorizontalScrollViews` para o Top e para os edifícios favoritos e ainda um `LinearLayout` para os serviços favoritos.

Neste *layout* específico é importante referir que a sua composição foi criada em XML mas a sua atualização é efetuada no próprio código da *Activity* devido ao seu dinamismo e condicionantes, isto é, quando o utilizador regista um ou vários edifícios como favoritos estas escolhas são refletidas com a presença dos favoritos na *TopActivity*. Para a representação de cada um dos edifícios favoritos foi criado um *layout* personalizado (ver Figura 4.8 – (a)) com um `ImageButton` cuja imagem é o logotipo do tipo de edifício em questão e uma `TextView` cujo texto é o nome do edifício. Para a representação de cada serviço favorito foi também criado um *layout* personalizado (ver Figura 4.8 – (b)) que consiste em três `TextViews` - (1) nome e (2) modalidade do serviço e (3) sigla da entidade responsável pelo serviço - contidas num `RelativeLayout`.

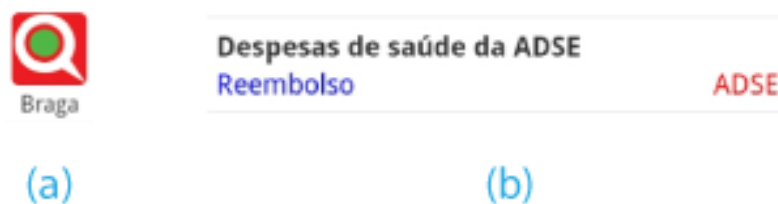


Figura: 4.8 – *Layouts* personalizados para os Favoritos (a) Loja | (b) Serviços

Tal como se pode observar pela Figura 4.6, a partir de um toque simples em qualquer item do Top 5 ou da lista de serviços favoritos o utilizador navega diretamente para a *Activity* responsável pelos detalhes do serviço em questão. Por sua vez, com o toque simples num item da lista de edifícios favoritos o utilizador navega para a lista de serviços filtrados pela loja em questão.

CategoriesActivity

A CategoriesActivity tem o intuito de apresentar a lista de categorias às quais os serviços estão associados. A apresentação do conteúdo neste caso é efetuada numa lista de registos, sendo esta Activity uma extensão da ListActivity.

Não sendo necessário qualquer *design* especial foram usados os *layouts* simples da própria *framework*. É de referir ainda que na composição do layout usado existe uma ListView na qual a Activity apresenta o seu conteúdo.

Um toque simples em qualquer elemento da lista de categorias invoca a Activity responsável pela lista de serviços, sendo estes filtrados pela categoria selecionada.

ServiceListActivity

No que diz respeito à ServiceListActivity, de forma análoga à Activity responsável pela lista de categorias, apresenta também os seus conteúdos numa lista estendendo a ListActivity. No entanto, neste caso existe ainda a possibilidade de se efetuar uma pesquisa entre os seus elementos através da introdução de um termo, através do qual o filtro será efetuado. Apesar de a ServiceListActivity ser o conteúdo de uma das Tabs da MainActivity, também pode ser invocada por outras Activities existindo nesse caso um filtro adicional (ver Figura 4.6).

O *layout* da ServiceListActivity, para além da ListView a ser preenchida pelos serviços possui ainda uma EditText na qual se pode escrever o termo que se pretenda utilizar para procurar pelo serviço desejado. Em relação à apresentação de cada item da lista, foi reutilizado o *layout* personalizado presente na Figura 4.8 – (b).

Um toque simples em qualquer item da lista de serviços permite que o utilizador navegue para a Activity responsável pelos detalhes do serviço selecionado. Já no caso de um toque prolongado, a ListActivity providencia um menu de contexto. Este menu tem como objetivo possibilitar várias funcionalidades referentes ao item premido, nomeadamente:

- **Ver Entidade** – navegar para a Activity responsável pelos detalhes da Entidade prestadora do serviço selecionado.
- **Ver Detalhes** – navegar para a Activity responsável pelos detalhes do serviço selecionado, da mesma forma que o toque simples.

- **Ver no Mapa** – navegar para a *Activity* responsável por mostrar no mapa os edifícios onde o serviço selecionado pode ser prestado.
- **Adicionar/Remover aos/dos Favoritos** – marca o serviço selecionado como favorito, adicionando o devido registo na BD. No caso de o serviço ser favorito é executada a ação inversa, ou seja, o registo adicionado previamente é removido da BD.

BuildingListActivity

A última *Activity* que estende a *ListActivity* é a *BuildingListActivity*, sendo esta responsável por apresentar a lista de edifícios onde os serviços são prestados.

O *layout* desta *Activity* é o mesmo que o da *CategoriesActivity*, usando apenas uma *ListView* para apresentar o seu conteúdo. No entanto, tal como na lista de serviços, foi criado um *layout* personalizado para apresentação de cada item da lista. Este *layout* (ver Figura 4.9) consiste de uma *ImageView* onde é apresentado o logotipo respeitante ao tipo de edifício em questão e duas *TextViews* - (1) nome e (2) número de telefone do edifício.



Figura: 4.9 – *Layout* personalizado de um item da lista de Edifícios

Um toque simples em qualquer elemento da lista de edifícios invoca a *Activity* responsável pela lista de serviços, sendo estes filtrados pelo edifício selecionado. Tal como na *ServiceListActivity*, também neste caso o toque prolongado despoleta um menu de contexto com as seguintes funcionalidades:

- **Ver Serviços** – navegar para a *Activity* responsável pela lista de serviços filtrados pelo edifício selecionado, tal como no toque simples.
- **Ver Detalhes** – navegar para a *Activity* responsável pelos detalhes do edifício selecionado.
- **Ver no Mapa** – navegar para a *Activity* responsável por mostrar no mapa o edifício selecionado.

- **Adicionar/Remover aos/dos Favoritos** – De forma análoga à mesma funcionalidade no menu de contexto na lista de serviços efetua as mesmas ações, mas neste caso para o edifício selecionado.

ContactActivity

A ContactActivity é responsável pela apresentação dos contactos disponíveis para suprir qualquer dúvida ou reportar qualquer inconsistência em relação à aplicação e/ou prestação de serviços público privados de uma forma geral.

EntityDetailsActivity

No caso da EntityDetailsActivity nenhuma funcionalidade especial é requerida, sendo esta criada pela extensão da classe Activity. É através desta Activity que são apresentados aos utilizadores os detalhes das Entidades prestadoras de serviços públicos e privados.

O *layout* da EntityDetailsActivity consiste de uma ImageView para a apresentação do logotipo da entidade em questão e TextViews onde são apresentados o resto dos detalhes. De notar apenas que a TextView responsável por mostrar o *Website* da entidade tem um atributo – *autoLink* - que permite direcionar o utilizador diretamente para esse mesmo Website através de uma aplicação própria para o efeito.

Esta Activity implementa um menu de opções com as funcionalidades que se seguem:

- **Ligar** – permite efetuar uma chamada telefónica para o número de contacto da Entidade, invocando a Activity de marcação pertencente à aplicação Telefone passando-lhe o respetivo número.
- **Email** – permite enviar uma mensagem de correio eletrónico para o endereço de contacto da Entidade expondo uma lista de aplicações para efeito, deixando ao critério do utilizador a escolha da aplicação a utilizar.
- **Portal do Cidadão** – direciona o utilizador para a página do Portal do Cidadão respeitante à entidade em questão, através de uma aplicação própria para o efeito.

BuildingDetailsActivity

A `BuildingDetailsActivity` é responsável pela apresentação dos detalhes dos edifícios, sendo por isso muito semelhante à `EntityDetailsActivity`. Os seus *layouts* são também muito semelhantes, consistindo então de uma `ImageView` para apresentar o logotipo respeitante ao tipo do edifício em questão e de `TextViews` para apresentar o resto dos detalhes.

No que diz respeito ao menu de opções desta *Activity*, este implementa praticamente as mesmas funções do menu de contexto da lista de edifícios. A funcionalidade de **Ver Detalhes** não é obviamente usada, havendo por sua vez, uma funcionalidade – **Ligar** - que permite que o utilizador efetue uma chamada telefónica para o edifício em questão de forma análoga à funcionalidade de mesmo nome na *Activity* dos detalhes das entidades.

ServiceDetailsActivity

A `ServiceDetailsActivity` é responsável pela apresentação dos detalhes dos serviços. No entanto, esta difere das outras *Activities* de detalhes porque no caso dos serviços a informação é de maior amplitude e está dividida em conteúdos. Estes conteúdos podem ser de vários tipos e cada um tem a sua apresentação. Posto isto, esta *Activity* estende a `TabActivity` cujos conteúdos das *Tabs* são a apresentação de cada tipo de conteúdo vinculado ao serviço em questão.

A *Activity* responsável pela apresentação de cada conteúdo – `WebViewActivity` – estende a classe *Activity* e possui apenas uma `WebView` para renderizar os conteúdos, já que os mesmos se encontram em HTML.

O *layout* da `ServiceDetailsActivity` é semelhante ao da `MainActivity`, fazendo uso do mesmo *layout* personalizado para as *tabs* exposto na Figura 4.7. Já o *layout* de apresentação de cada conteúdo é todo tratado pela `WebView` de acordo com as *Tags* HTML utilizadas.

O menu de opções implementado pela *Activity* responsável pelos detalhes dos serviços implementa praticamente as mesmas funcionalidades que o menu de contexto da `ServiceListActivity`. Neste caso a funcionalidade **Ver Detalhes** não é implementa-

da, havendo uma opção que direciona o utilizador para à página do Portal do Cidadão referente ao serviço em questão.

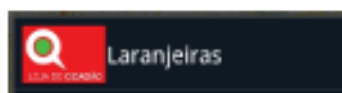
BuildingsMapActivity

Por fim, a `BuildingsMapActivity` é responsável pela apresentação dos edifícios com recurso aos mapas da Google. Quando a Activity é invocada esta pode conter dois filtros distintos - (1) por Edifício e (2) por Serviço. No primeiro caso, o mapa apenas apresenta o edifício pretendido coincidindo este com o centro do mapa. Em caso do filtro ter proveniência numa das Activities responsáveis pelos serviços, são mostrados todos os edifícios onde o serviço selecionado é prestado e o mapa centra-se na posição atual do utilizador fornecida pelo próprio dispositivo. De referir ainda que para a utilização desta Activity é necessário ter acesso à *Internet* e fazer uso de uma chave própria fornecida pela Google.

O *layout* da `BuildingsMapActivity` usa uma `MapView` para que o mapa possa ser visualizado. Neste caso concreto os edifícios apresentados no mapa usam uma imagem padrão do Android que pode ser observada na Figura 4.10 - (a). Foi também criado um *layout* personalizado para o título do diálogo mostrado quando um edifício é selecionado (ver Figura 4.10 - (b)). Este título consiste em uma `ImageView` com a imagem do tipo de edifício em questão e uma `TextView` com o seu nome.



(a)



(b)

Figura: 4.10 – (a) Marcador de um Edifício no Mapa | (b) *Layout* personalizado do Título do Diálogo

Um toque simples no marcador que representa um edifício despoleta o aparecimento de um diálogo com dois botões. Estes botões permitem efetuar duas funcionalidades relacionadas com o edifício selecionado – (1) Navegar e (2) Ligar. A primeira permite que o utilizador use as coordenadas do edifício para obter uma rota na aplicação *Navigation* e a segunda permite que o utilizador efetue uma chamada telefónica

para o edifício em questão, da mesma forma que as Activities responsáveis pelos edifícios.

A própria MapView lida com a navegação pelo mapa através de um efeito de “arraste”. São ainda providenciados mecanismos de *zoom*, quer através de botões próprios para o efeito, quer através do efeito “*pintch to zoom*”³.

O menu de opções implementado pela MapActivity providencia três funcionalidades relacionadas com o mapa:

- **Minha Localização** – Posiciona o mapa com o centro na localização atual fornecida pelo dispositivo.
- **Satélite** – Altera a visualização do mapa para a vista de satélite (ver Figura 4.11 - (a)), caso se encontre na vista Mapa.
- **Vista Normal** – Altera a visualização para a vista Mapa (ver Figura 4.11 - (b)) caso se encontre na vista Satélite.

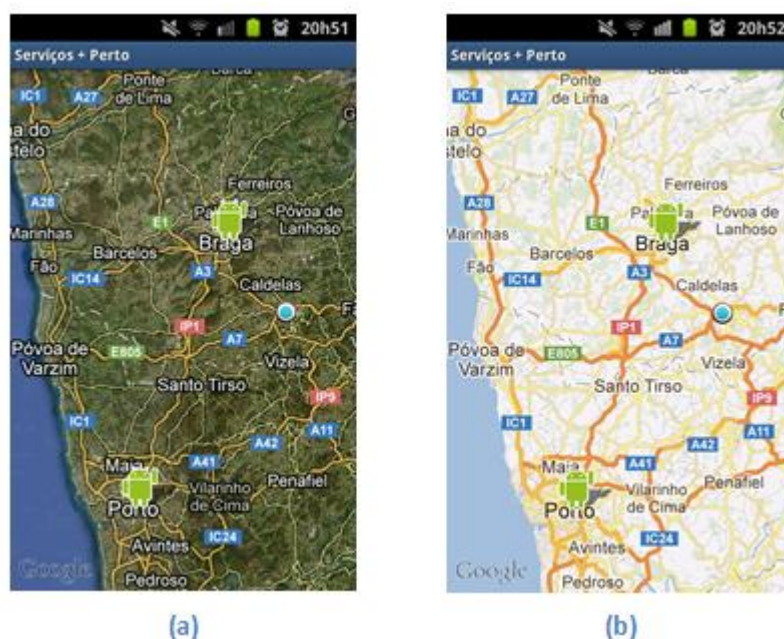


Figura: 4.11 – (a) Vista de Satélite | (b) Vista Normal

³ Termo usado para descrever um movimento específico dos dedos utilizado para aumentar/reduzir o *zoom*, permitido por dispositivos *mulyi-touch*.

4.3.3. *Permissões*

Originalmente, nenhuma aplicação Android tem permissão para efetuar qualquer operação que possa ter “algum impacto adverso noutras aplicações, no sistema operativo, ou para o utilizador” (22). Para que uma aplicação possa ter acesso a informação que seja privada, como os dados dos contactos ou de outras aplicações, assim como acesso a funcionalidades protegidas dos dispositivos, necessita de permissões específicas para o efeito. Posto isto, a aplicação necessita de 5 (cinco) permissões:

- ***Access_fine_location*** – Permite que uma aplicação tenha acesso à localização do dispositivo de uma forma precisa (por exemplo, GPS).
- ***Access_coarse_location*** – Permite que uma aplicação tenha acesso à localização do dispositivo de uma forma não muito precisa (por exemplo, Wi-Fi).
- ***Internet*** – Permite que uma aplicação tenha acesso à internet.
- ***Call_Phone*** – Permite que uma aplicação inicie uma chamada telefónica sem que para isso tenha de passar pela marcação.
- ***Call_Privileged*** – *Permite que uma aplicação efetue uma chamada telefónica para qualquer telemóvel, incluindo chamadas de emergência, sem que para isso tenha de passar pela marcação.*

5. Caso Prático de Utilização

No presente capítulo vai ser demonstrado um caso prático com a finalidade de apresentar o sistema criado, tanto na componente de *back-office* como na aplicação Android. Deste modo, vai ser exposta uma situação que envolve a introdução de informação na base de dados do servidor e outra que pretende simular a utilização da aplicação por um cidadão, de modo a que seja dado uso à informação introduzida.

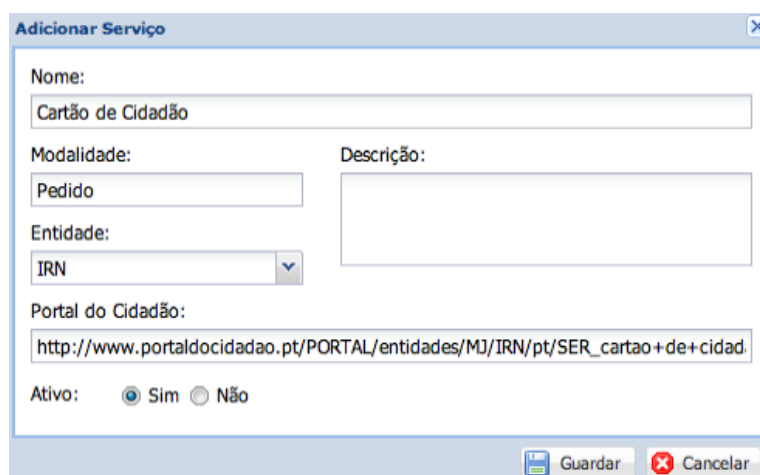
5.1. *Fluxo da Informação*

Para que seja possível demonstrar o processo de inserção e atualização da informação no sistema por parte de um agente com essa responsabilidade, será introduzido um conjunto de dados na BD do servidor. Os dados a introduzir necessitam de compor todo o sistema estrutural da prestação de um serviço por parte de uma entidade. Tendo em conta a sua importância e utilização por parte dos cidadãos, o serviço a ser demonstrado será o Cartão do Cidadão.

Quando se pretende introduzir um serviço na BD a primeira verificação a ser efetuada é se entidade responsável pelo serviço em questão já existe no sistema, dado que apenas com a sua presença é possível introduzir o serviço e efetuar o devido vínculo. A inserção de uma entidade pode ser realizada através de um formulário próprio existente para o efeito, tal como se pode verificar na Figura 5.1. Para isso, quando o agente se depara com o *back office* pode efetuar um clique na opção **Entidades** do menu de navegação que se situa do lado esquerdo. De seguida a lista das entidades surge no painel central e, caso a entidade desejada ainda não se encontre no sistema, deve proceder à sua inserção. Na eventualidade de ser necessário editar ou eliminar as informações inerentes a uma entidade o procedimento é idêntico. De salientar apenas que para se proceder à eliminação de uma entidade, esta não pode ter serviços ou balcões a si vinculados.

Figura: 5.1 – Janela de Inserção/Edição de Entidades

A presença do IRN no sistema torna possível a inserção de qualquer serviço prestado por esta entidade pública. Deste modo, através da janela de introdução de um novo serviço podem ser adicionadas todas as informações relativas ao cartão de cidadão (ver Figura 5.2) e efetuado o devido vínculo ao IRN através da lista de entidades presentes neste formulário. Tal como as entidades também os serviços podem ser editados e/ou eliminados. No entanto, para que os serviços possam ser eliminados não se pode verificar a existência de qualquer vínculo por parte destes com nenhum balcão, categoria ou conteúdos.



Adicionar Serviço

Nome:

Modalidade: Descrição:

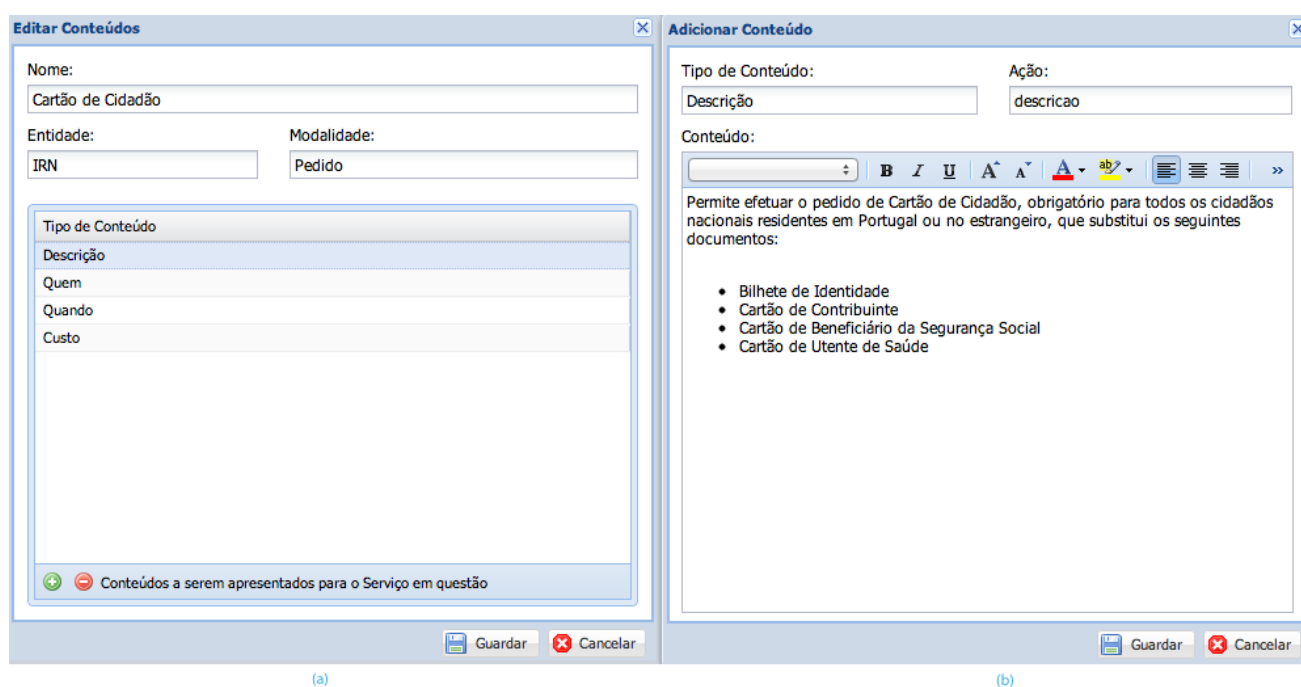
Entidade:

Portal do Cidadão:

Ativo: ☒ Sim ☐ Não

Figura: 5.2 – Janela de Inserção/Edição de Serviços

Depois de efetuada a introdução do cartão de cidadão de modalidade pedido, pode agora proceder-se à adição dos conteúdos respeitantes a este serviço. Neste caso prático serão usados os 4 (quatro) tipos de conteúdos: Descrição, Quem, Quando e Custo. Como se pode verificar pela Figura 5.3 - (a), existe uma janela onde se pode visualizar a lista de conteúdos associados a cada serviço. Adicionalmente, para que se possam introduzir/editar os conteúdos de um serviço de forma simples e intuitiva, existe uma janela com um editor HTML WYSIWYG (do inglês *What You See Is What You Get*). Desta forma, no momento da introdução dos conteúdos é possível ter uma noção de como estes serão apresentados na aplicação *mobile* (ver Figura 5.3 - (b)).



Editar Conteúdos

Nome:

Entidade: Modalidade:

Tipo de Conteúdo

- Descrição
- Quem
- Quando
- Custo

Adicionar Conteúdo

Tipo de Conteúdo: Ação:

Conteúdo:

Permite efetuar o pedido de Cartão de Cidadão, obrigatório para todos os cidadãos nacionais residentes em Portugal ou no estrangeiro, que substitui os seguintes documentos:

- Bilhete de Identidade
- Cartão de Contribuinte
- Cartão de Beneficiário da Segurança Social
- Cartão de Utente de Saúde

Figura: 5.3 – (a) Janela de vínculo entre Conteúdos e Serviços | (b) Janela de Inserção/Edição de Conteúdos

A introdução de novas categorias pode ser obtida da mesma forma que as descritas anteriormente para a introdução de entidades e serviços. No entanto, no sistema já existem 14 (catorze) categorias que foram obtidas através do Portal do Cidadão e do portal Senha001. Posto isto, apenas será efetuado o vínculo a uma categoria já existente como se pode observar pela Figura 5.4.

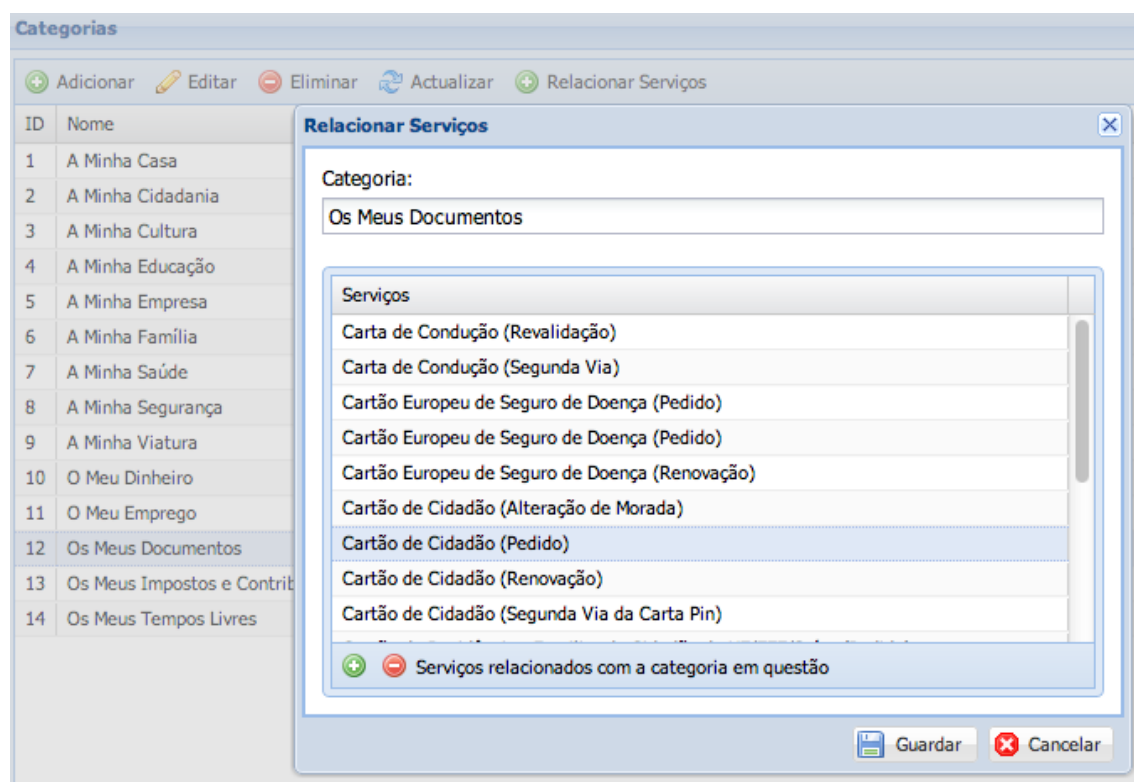


Figura: 5.4 – Janela de vínculo entre Categorias e Serviços

Neste momento é necessário introduzir a panóplia de locais onde o utilizador pode efetuar o pedido do seu cartão de cidadão. Como tal, é necessária a existência do(s) tipo(s) de edifício(s) que categorizem os diferentes edifícios a introduzir. Neste caso concreto, serão introduzidas praticamente todas as lojas do cidadão, não sendo efetuado contudo, o vínculo com todas elas dado o objetivo de testar o comportamento da aplicação. A inserção, edição e eliminação de Tipos de Edifício, Edifícios, Balcões e Relações entre Balcões e Serviços são realizadas de forma idêntica à dos registos já descritos. Na Figura 5.5 pode ser verificado um exemplo de um vínculo criado entre o balcão do IRN na LC do Porto e o serviço responsável pelo pedido do cartão de cidadão.

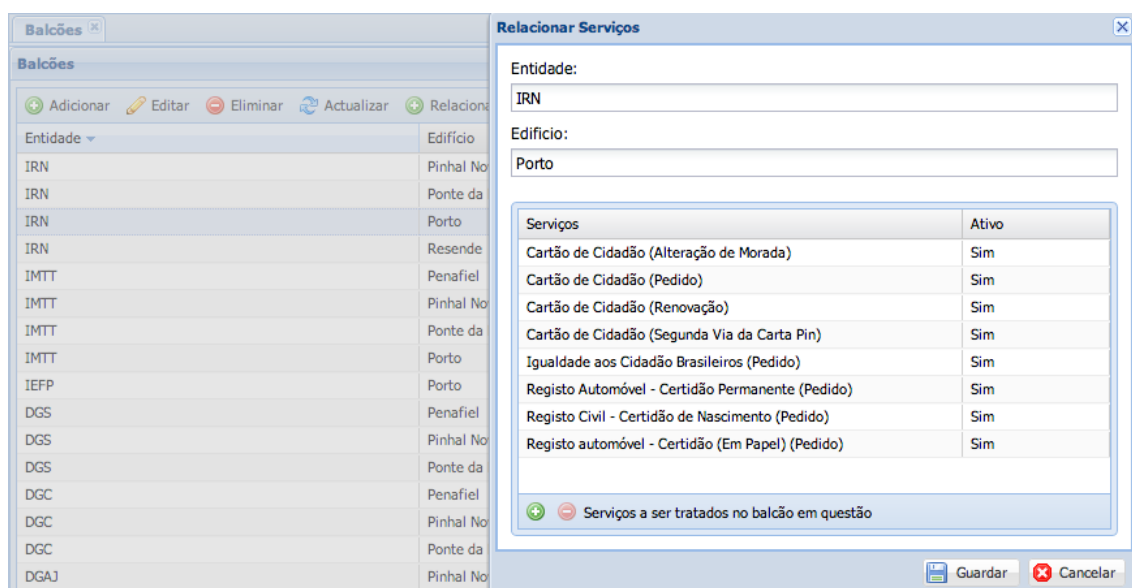


Figura: 5.5 – Janela de vínculo entre Balcões e Serviços

5.2. *Uso da Aplicação Android*

No seguimento do caso de estudo irão ser demonstradas as potencialidades inerentes à solução Android criada. Para tal, serão percorridos os fluxos adjacentes às pretensões de um cidadão cujo objetivo passa pela obtenção de informações auxiliares à realização do serviço referido.

É de salientar ainda que foram também introduzidas informações relativas a outras entidades e serviços, com o intuito de apresentar uma maior composição da aplicação no teste a efetuar.

Quando um utilizador inicia pela primeira vez a aplicação posteriormente à sua instalação, depara-se com a vista do Top e dos Favoritos, estando estes últimos desprovidos de qualquer registo (ver Figura 5.6).



Figura: 5.6 – Página de entrada na Aplicação (Sem Favoritos)

Tendo em conta que o pedido do cartão do cidadão é o serviço mais utilizado por parte dos cidadãos portugueses, este consta no Top 5 (cinco) em primeiro lugar, como se pode observar pela Figura 5.6. Através do toque na imagem do cartão do cidadão o utilizador seria apresentado com os detalhes do serviço. Porém, nesta demonstração serão explorados outros meios para atingir esses mesmos detalhes.

As outras formas que possibilitam alcançar as informações de qualquer serviço passam pelas duas hipóteses de navegação, ou então diretamente através da lista de todos os serviços existentes. O primeiro tipo de navegação (ver Figura 5.7 - (a)) pode ser realizado através da lista de categorias, ou seja, com a escolha de uma categoria contida na lista apresentada, o utilizador pode alcançar todos os serviços que possuem um vínculo com a mesma. De forma análoga, se o utilizador optar por navegar através da lista dos edifícios poderá aceder a todos os serviços que são vinculados a balcões presentes nesses.

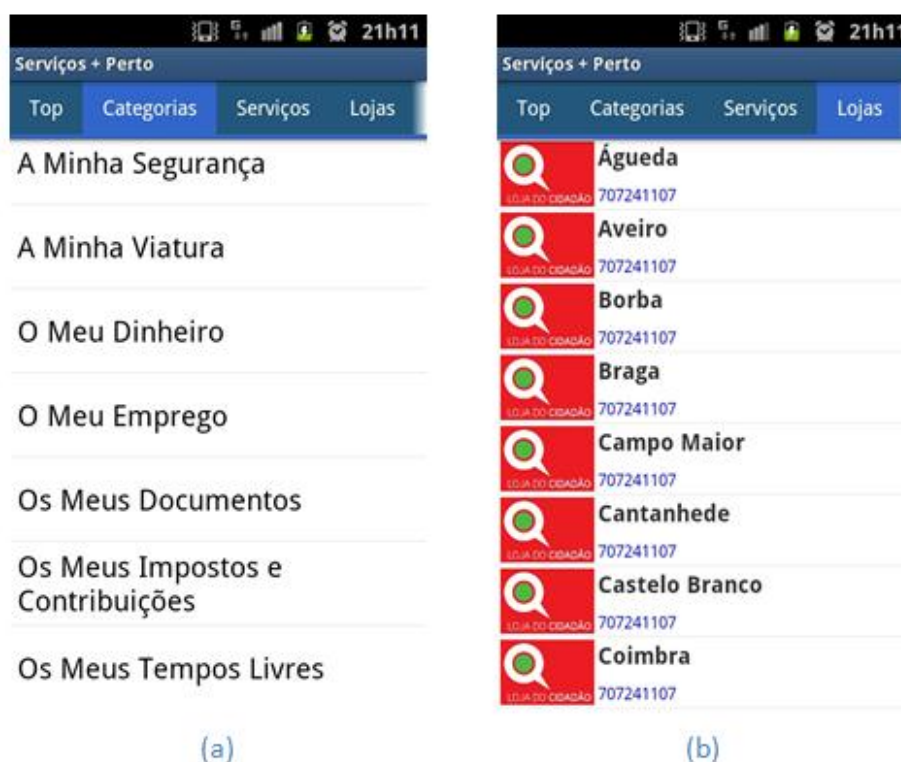


Figura: 5.7 – (a) Lista de Categorias | (b) Lista de Edifícios

Caso o utilizador deseje pesquisar pelo serviço que pretende realizar de uma forma mais concisa, pode alcançar diretamente a lista de todos os serviços existentes. De notar que a lista a ser apresentada ao cidadão, independentemente do meio utilizado para a alcançar, possui as mesmas funcionalidades. Como se pode observar pela Figura 5.7 - (b), a lista de serviços é dotada de um sistema de pesquisas através do qual se pode obter o serviço pretendido de uma forma mais rápida.

Para que se possa verificar os resultados práticos da informação introduzida no back office, é efetuada uma navegação pela categoria **Os Meus Documentos** obtendo a lista presente na Figura 5.8 - (a), estando o cartão de cidadão presente. Da mesma forma, na Figura 5.9 - (b) pode-se observar a presença do cartão de cidadão numa navegação através da lista dos edifícios, tendo sido selecionada a loja do cidadão do **Porto**.



Figura: 5.8 – (a) Lista de Serviços por Categorias | (b) Lista de Serviços por Edifícios

Depois de alcançado o serviço pretendido, o utilizador pode obter os seus detalhes com o toque simples sobre o elemento da lista, ou então através do menu de contexto que surge com o toque prolongado no mesmo elemento. Posto isto, pode-se verificar através da Figura 5.9 os conteúdos introduzidos anteriormente, sendo a sua apresentação idêntica à que se podia observar no momento da inserção/edição (ver Figura 5.3). Adicionalmente aos conteúdos presentes, existe ainda um menu de opções relacionadas com o cartão de cidadão.

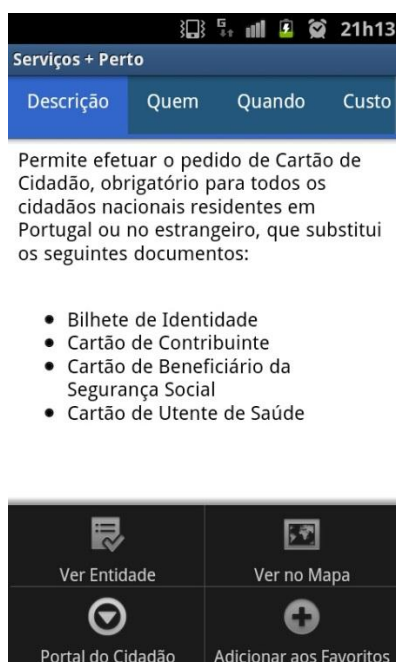


Figura: 5.9 – Detalhes do Serviço Cartão de Cidadão

Depois de uma leitura atenta às informações relativas ao cartão de cidadão o utilizador pode proceder à verificação do local mais próximo onde pode realizar este serviço através da opção **Ver no Mapa**. A Figura 5.10 permite constatar o aparecimento de todos os edifícios que contêm os balcões, em que existe um vínculo ao cartão de cidadão. Este mapa poderia ser obtido também através da opção análoga do menu de contexto presente na lista de serviços.



Figura: 5.10 – Mapa com os Edifícios que permitam efetuar o Pedido do Cartão de Cidadão

Nesta fase, o utilizador poderá optar pelo edifício que melhor se adequar às suas necessidades e obter uma rota de navegação (ver Figura 5.11), ou então efetuar uma chamada telefónica caso pretenda alguma informação adicional.

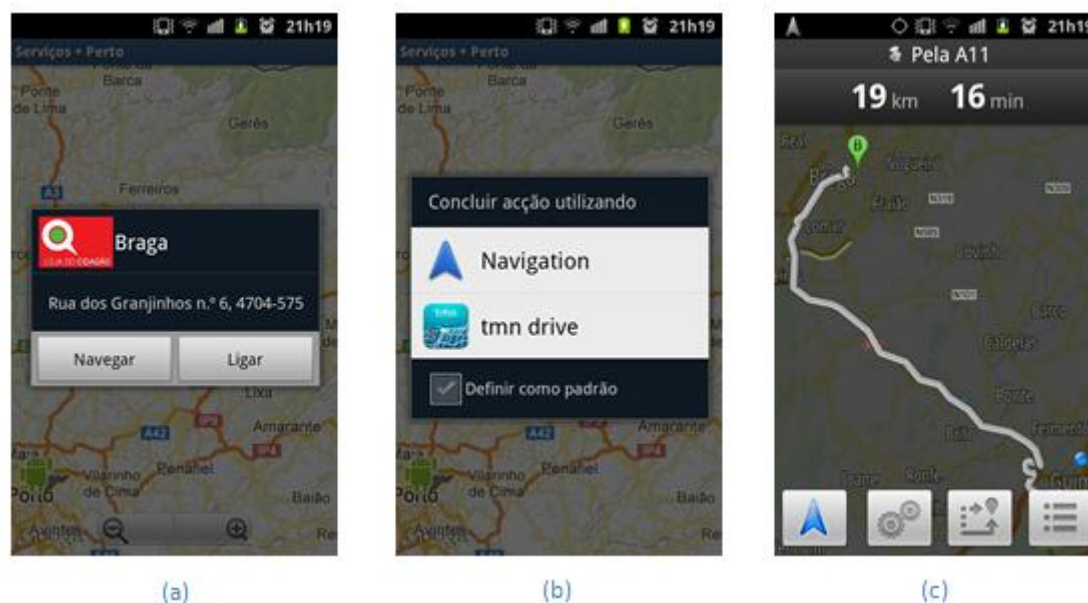


Figura 5.11 - (a) Diálogo de Edifício | (b) Escolha de Navegação | (c) Rota no Navigation

No caso de o utilizador pretender obter no mapa a localização de um edifício específico, o processo seria semelhante ao efetuado anteriormente, ou seja, na lista dos edifícios teria de seleccionar a opção **Ver no Mapa** existente no menu de contexto, exposto pelo toque prolongado num dos edifícios, ou ainda, através da opção análoga do menu de opções presente nos detalhes do edifício em questão. Dessa forma, o mapa apresentado apenas conteria o edifício pretendido.



Figura: 5.12 – Página de Entrada (Com Favoritos)

Por fim, durante o caso de estudo procedeu-se à marcação como favorito do serviço **Cartão de Cidadão** e das lojas do cidadão de **Braga** e das **Laranjeiras**. O resultado destas ações pode ser observado na Figura 5.12.

6. Conclusões e Trabalho Futuro

6.1. Conclusões

Este trabalho de dissertação focou-se no estudo de formas de apresentar aos cidadãos informações de uma forma simples e muito objectiva sobre os serviços públicos e privados nas Lojas do Cidadão. Do estudo das informações existentes, verificou-se que através de uma boa organização das mesmas aliada às potencialidades dos dispositivos móveis é possível criar soluções capazes de auxiliar o cidadão no seu quotidiano.

A solução alcançada resolve o problema do tratamento e preparação dos dados, através de uma plataforma *web* capaz de centralizar informações dispersas por vários meios, para que depois esta possa ser utilizada numa aplicação para Android. Reforça-se que as especificidades dos dispositivos móveis obrigam à edição dos conteúdos para os mesmos. A descrição de um mesmo serviço pode ser feita com muito texto e muitos links no portal do cidadão, mas essa quantidade de texto não pode ser apresentada num único ecrã do dispositivo móvel. A informação teve de ser reorganizada e editada especificamente para o novo cliente móvel.

As plataformas de desenvolvimento escolhidas, tanto para o desenvolvimento sobre a Web, assim como para a aplicação mobile, mostraram-se adequadas para a criação de todo o sistema envolvido na solução pretendida. A documentação de referência que é disponibilizada como auxílio à programação com as *frameworks* utilizadas foi uma mais-valia, reduzindo o tempo de aprendizagem necessário para trabalhar com as mesmas.

A componente de *back-office*, responsável pela introdução e atualização da base de dados oferece os mecanismos necessários, de uma forma agradável, não sendo necessários conhecimentos informáticos aprofundados para que a informação seja mantida com a frequência necessária. Tendo em conta que esta componente foi criada para funcionar sobre a Web, a sua utilização não está dependente de nenhum sistema operativo em específico.

A aplicação para Android é capaz de apresentar um vasto leque de funcionalidades, que permitem tornar as informações relacionadas com toda a estrutura de prestação de serviços públicos e privados em mais-valias para os cidadãos. Através desta, é possível preparar melhor a realização de um serviço, sabendo de antemão os requisitos os horários e os locais. A aplicação móvel permite contactar diretamente as entidades públicas e privadas responsáveis pelos serviços, através de uma chamada telefónica ou do envio de um email. Caso seja pretendido, é ainda possível a obtenção de uma rota de navegação para o balcão que se defina como mais adequado à realização de qualquer serviço que conste das necessidades de qualquer cidadão.

É importante salientar que, apesar das ótimas indicações deixadas pelos testes efetuados, seria fundamental a existência de uma versão consistente da base de dados para que a aplicação pudesse ser disponibilizada a todos os cidadãos. Através das respostas por parte destes, tornar-se-ia possível realizar uma melhor avaliação do real valor da solução criada.

Por fim, todo o trabalho desenvolvido será disponibilizado na internet com o intuito de agilizar o processo de avaliação e melhoria do mesmo, assim como de motivar a produção de soluções enquadradas no mesmo âmbito.

6.2. *Trabalho Futuro*

Este projeto teve origem numa parceria entre a Universidade do Minho e a AMA que visa a criação de soluções no âmbito da disponibilização de informação aos cidadãos, de forma para que estes usufruam de um conhecimento sustentado em relação ao meio que os rodeia. Posto isto, foi criada uma solução que serve, numa primeira instância, de extensão ao portal do cidadão.

Tendo em conta as potencialidades dos dispositivos móveis, que se encontram em grande evolução e massificação, tornou-se possível desenvolver soluções capazes de elevar e agilizar todo o processo inerente à prestação dos serviços públicos e privados.

No que diz respeito ao alcance da solução, os próximos passos passam por uma adaptação da aplicação mobile aos restantes sistemas operativos, com especial ênfase

ao iOS e ao WindowsPhone, que em conjunto com o Android abrangem a maior parte dos utilizadores de smartphones e tablets.

Relativamente às informações, é importante que se proceda à conclusão de uma base de dados consistente para possibilitar o lançamento da aplicação Android, o mais breve quanto possível. Tal disponibilização iria permitir uma resposta por parte dos utilizadores, constituindo esta uma primeira avaliação real do impacto da aplicação.

O processo de atualização da informação na aplicação, excetuando a componente de notícias, implica uma nova versão da aplicação a ser lançada. Uma possível evolução deste processo deverá contemplar uma atualização por parte da própria aplicação. Para tal, deverá haver o conceito de versão de BD no servidor que poderá ser verificada através da internet.

Os mapas da Google são, neste momento, a ferramenta que proporciona um melhor desempenho na integração com as aplicações para Android. Contudo, existem alternativas que se encontram em constante evolução e que permitem o seu uso sem a necessidade de acesso à internet, como o OSM (de Open Street Maps) (23).

Em relação a informações adicionais que poderão representar mais-valias para a evolução da solução criada, podem passar por estatísticas sobre locais de estacionamento, tempos de atendimento, horários de prestação dos serviços nos balcões, assim como pela integração com os horários dos transportes públicos.

Quanto às funcionalidades da aplicação para os dispositivos móveis, seria determinante que no futuro o cidadão pudesse realizar serviços que não necessitam de atendimento presencial através da mesma.

Referências Bibliográficas

1. **Shailendra C. Jain Palvia, Sushil S.Sharma**, “*E.Government and E-Governance: Definitions/Domain Framework and Status around the World*”, 5th International Conference on E-governance (ICEG), 2007.
2. **Jeff Breen**, “At the Dawn of E-Government: The Citizen as Costumer”, Government Finance Review, 2000.
3. **Department of Economic and Social Affairs – United Nations**, “E-Government Survey 2012: E-Government for the People”, 2012.
4. AMA. [Online] <http://www.ama.pt>.
5. OCDE, “Tornar a Vida Mais Fácil para Cidadãos e Empresas em Portugal: Administração Electrónica e Simplificação”, Cap.1 – Perfil e Contexto do País, pp. 25-32, 2009.
6. Apps for Democracy. [Online] <http://www.appsfordemocracy.org/about>.
7. Addam DuVander, “Government Data Contests Abound: SF, NYC and Australia”, ProgrammableWeb, 2009. [Online] <http://blog.programmableweb.com/2009/10/14/government-data-contests-abound-sf-nyc-and-australia>.
8. CivicApps. [Online] <http://www.civicapps.org>.
9. AppsForItaly. [Online] <http://www.appsforitaly.org/en/blog/comunicato>.
10. BarcelonaDigital. [Online] <http://www.bdigital.org/en/servicios/Pages/DetallesAgenda.aspx?xid=192>.
11. Chicago Tribune. [Online] http://articles.chicagotribune.com/2011-12-16/business/chi-apps-for-metro-chicago-winner-named-20111216_1_android-app-top-prize-grand-prize.
12. Mi Parque. [Online] <http://miparquelv.wordpress.com>.
13. WhereDoesMyMoneyGoes. [Online] <http://wheredoesmymoneygo.org/about.html>.
14. **C. Batini, S. Ceri, S. Kant and B. Navathe**, “Conceptual Database Design: An Entity Relational Approach”, 1991.
15. **C. J. Date**, “An Introduction to Database Systems”, 5th ed., 1990.

16. **Candace C. Fleming and Barbara von Halle**, “Handbook of Relational Database Design”, 1989.
17. Android Developer Adt. [Online]
<http://developer.android.com/sdk/installing/installing-adt.html>.
18. **Loiane Groner**, “Ext JS 4 First Look”, Chapter1: What’s New in Ext JS 4?, pp. 7-39, 2011.
19. **Jesus Garcia, Jacob K. Andersen and Grgur Grisogono**, “Ext JS in Action”, 2nd ed., Part1: Introduction to Ext JS 4, 2011.
20. **Zifurd Mednicks, Laird Dornin, G. Blake Meike and Masumi Nakamura**, “Programming Android”, O’Reilly, 2011.
21. **Satya Komatineni and Dave McLean**, “Pro Android 4”, Apress, 2012
22. Android Developer Permissions. [Online]
<http://developer.android.com/guide/topics/security/permissions.html>.
23. OpenStreetMap. [Online] <http://www.openstreetmap.org/.w.openstreetmap.org/>.
24. SpotHero. [Online] <https://itunes.apple.com/us/app/spothero/id499097243?mt=8>.
25. Fixit! [Online]
<https://play.google.com/store/apps/details?id=edu.northwestern.cs.aqualab.android.Fixit>.
26. Taxi share – Chicago. [Online]
<https://play.google.com/store/apps/details?id=com.chicagoandroid.taxishare>.